



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**TUGAS AKHIR - K141502**

# **RANCANG BANGUN PERANGKAT LUNAK PENDETEKSI PERINTAH SUARA BERBAHASA INDONESIA BERBASIS ANDROID UNTUK MENGOPERASIKAN KOMPUTER STUDI KASUS PEMBELAJARAN DI KELAS**

**MOHAMMAD IQBAL RAMADHAN**  
NRP 5112 100 076

Dosen Pembimbing  
Dwi Sunaryono, S.Kom., M.Kom.  
Nurul Fajrin A., S.Kom., M.Sc.

**JURUSAN TEKNIK INFORMATIKA**  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*



**TUGAS AKHIR - K141502**

# **RANCANG BANGUN PERANGKAT LUNAK PENDETEKSI PERINTAH SUARA BERBAHASA INDONESIA BERBASIS ANDROID UNTUK MENGOPERASIKAN KOMPUTER STUDI KASUS PEMBELAJARAN DI KELAS**

**MOHAMMAD IQBAL RAMADHAN**  
NRP 5112 100 076

Dosen Pembimbing  
Dwi Sunaryono, S.Kom., M.Kom.  
Nurul Fajrin A., S.Kom., M.Sc.

**JURUSAN TEKNIK INFORMATIKA**  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*



**FINAL PROJECT - K141502**

# **DESIGN AND IMPLEMENTATION OF OPERATING COMPUTER USING VOICE COMMAND IN BAHASA INDONESIA ON ANDROID PLATFORM CASE STUDY IN CLASS LEARNING**

**MOHAMMAD IQBAL RAMADHAN**  
NRP 5112 100 076

Advisor  
Dwi Sunaryono, S.Kom., M.Kom.  
Nurul Fajrin A., S.Kom., M.Sc.

**INFORMATICS DEPARTMENT**  
Faculty of Information Technology  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017

*[Halaman ini sengaja dikosongkan]*

## **LEMBAR PENGESAHAN**

# **RANCANG BANGUN PERANGKAT LUNAK PENDETEKSI PERINTAH SUARA BERBAHASA INDONESIA BERBASIS ANDROID UNTUK MENGOPERASIKAN KOMPUTER STUDI KASUS PEMBELAJARAN DI KELAS**

## **TUGAS AKHIR**

Diajukan Guna Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Bidang Studi Manajemen Informasi  
Program Studi S-1 Jurusan Teknik Informatika  
Fakultas Teknologi Informasi  
Institut Teknologi Sepuluh Nopember

Oleh :

**MOHAMMAD IQBAL RAMADHAN**

NRP : 5112 100 076

Disetujui oleh Dosen Pembimbing Tugas Akhir

Dwi Sunaryono, S.Kom., M.Kom.

NIP: 19720528 199702 1 001

(pembimbing 1)

Nurul Fajrin A., S.Kom., M.Sc.

NIP: 19860722 201504 2 003

DEPARTEMEN.....  
TEKNIK INFORMATIKA (pembimbing 2)

**SURABAYA  
MEI 2017**

*[Halaman ini sengaja dikosongkan]*



# **RANCANG BANGUN PERANGKAT LUNAK PENDETEKSI PERINTAH SUARA BERBAHASA INDONESIA BERBASIS ANDROID UNTUK MENGOPERASIKAN KOMPUTER STUDI KASUS PEMBELAJARAN DI KELAS**

Nama Mahasiswa : Mohammad Iqbal Ramadhan  
NRP : 5112 100 076  
Jurusan : Teknik Informatika FTIf-ITS  
Dosen Pembimbing 1 : Dwi Sunaryono, S.Kom., M.Kom.  
Dosen Pembimbing 2 : Nurul Fajrin A., S.Kom., M.Kom.

## **ABSTRAK**

*Pengembangan aplikasi pengendali jarak jauh sangat dibutuhkan saat ini. Penggunaan perangkat yang dalam operasionalnya tidak boleh menggunakan tangan membuat aplikasi pengendali jarak jauh menjadi keunggulan tersendiri. Dengan berbagi jaringan yang sama, dua perangkat atau lebih, dapat dikendalikan oleh pengguna untuk suatu kebutuhan tertentu. Pengendalian ini bentuknya bermacam-macam, salah satunya adalah pengendalian perangkat dengan suara.*

*Pengenalan suara-ke-teks adalah proses menerjemahkan gelombang suara hasil pengucapan kata-kata menjadi bentuk teks yang sesuai. Saat ini, pengenalan suara-ke-teks telah banyak diterapkan pada perangkat ponsel cerdas. Pengenalan suara-ke-teks banyak diterapkan untuk menulis keluaran suara menjadi teks atau menjalankan perintah tertentu tanpa harus mengetikkan sendiri isi teks atau menekan tombol yang bersesuaian. Dengan menggunakan teknologi ini, manusia dengan ponsel cerdas miliknya, bisa memberi perintah suara yang diubah menjadi teks terhadap*

*perangkat lain yang tersambung dalam suatu jaringan. Atas dasar itulah, muncul ide untuk menggabungkan keduanya dengan membangun aplikasi pengenalan suara untuk mengoperasikan komputer berbasis Android menggunakan Bahasa Indonesia.*

*Teknologi pengenalan suara-ke-teks yang digunakan pada aplikasi ini menggunakan API masukan yang disediakan oleh Google. Respon yang diberikan aplikasi pada pengguna yaitu berupa berjalannya perintah pengoperasian komputer yang diucapkan oleh pengguna. Perintah suara yang diberikan adalah membuka aplikasi, menjalankan aplikasi, dan operasi mematikan komputer.*

*Dari hasil pengujian yang ada pada Tugas Akhir ini, pengguna dapat mengoperasikan aplikasi powerpoint pada komputer menggunakan aplikasi ini dengan bantuan ponsel cerdas sebagai media pengenalan suaranya. Aplikasi powerpoint berjalan dengan baik sesuai dengan perintah yang diberikan. Teknologi pengenalan suara-ke-teks pada aplikasi ini juga berjalan dengan cukup baik.*

*Dari pengujian aplikasi ini dapat diambil kesimpulan bahwa aplikasi ini dapat membantu pengguna dalam mengoperasikan aplikasi powerpoint pada komputer menggunakan ponsel cerdas berbasis android. Pengembangan lebih lanjut mengenai teknologi suara-ke-teks sangat diperlukan untuk mengembangkan aplikasi in sehingga perintah yang dijalankan tidak hanya untuk mengoperasikan powerpoint saja.*

**Kata kunci:** Akses Jarak Jauh, Pengendali Jarak Jauh, Pengenalan Suara-ke-teks, Android, Pengoperasian PowerPoint

# DESIGN AND IMPLEMENTATION OF OPERATING COMPUTER USING VOICE COMMAND IN BAHASA INDONESIA ON ANDROID PLATFORM CASE STUDY IN CLASS LEARNING

Name : Mohammad Iqbal Ramadhan  
NRP : 5112 100 076  
Major : Teknik Informatika FTIf-ITS  
Supervisor I : Dwi Sunaryono, S.Kom., M.Kom.  
Supervisor II : Nurul Fajrin A., S.Kom., M.Sc.

## ABSTRACT

*Remote control application development is needed today. The use of devices that are in operational use should not use hands to make remote control applications into their own advantages. By sharing the same network, two or more devices, can be controlled by the user for a particular need. This control of various forms, one of which is controlling the device with voice.*

*Voice-to-text recognition is the process of translating sound waves of pronunciation of words into appropriate text form. Currently, voice-to-text recognition has been widely applied to smartphone devices. Voice-to-text recognition applies to writing voice output to text or running certain commands without having to type in the text itself or press the corresponding key. By using this technology people with their smartphones, can give text commands to other devices connected in a network. On that basis, came the idea to combine the two by building a voice recognition app to operate computer on an Android-based using Bahasa Indonesia.*

*The voice-to-text recognition technology used in this app uses the input API provided by Google. Response given the application on the user is a running computer operation command spoken by the user. Voice commands provided are open application, operate application, and shutdown computer.*

*From the result of existing tests on this Final Project, users can operate powerpoint application on the computer using this application with the help of smart phones as voice recognition media. Powerpoint app works well according to the given command. The voice-to-text recognition technology in this app also works pretty well.*

*From testing this application can be concluded that this application can help users in operating powerpoint application on computer using android based smartphone. Further development of voice-to-text technology is essential for developing applications in so that commands are run not only to operate powerpoints only.*

***Keyword: Remote Access, Remote Control, Voice-to-text Recognition, Android, Operating PowerPoint***

## **KATA PENGANTAR**

Alhamdulillahirobbil ‘alamiin, puji syukur kepada Tuhan Yang Maha Esa atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul:

### **RANCAN BANGUN PERANGKAT LUNAK PENDETEKSI PERINTAH SUARA BERBAHASA INDONESIA BERBASIS ANDROID UNTUK MENGOPERASIKAN KOMPUTER STUDI KASUS PEMBELAJARAN DALAM KELAS**

Melalui lembar ini, penulis ingin menyampaikan ucapan terimakasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah SWT atas segala nikmat dan rahmat yang telah diberikan selama ini.
2. Ayah, Ibu dan keluarga penulis yang tiada henti- hentinya mencurahkan kasih sayang, perhatian dan doa kepada penulis selama ini.
3. Bapak Dwi Sunaryono selaku dosen pembimbing yang telah memberikan bimbingan, motivasi, nasihat dan meluangkan waktu untuk membantu pengerjaan Tugas Akhir ini.
4. Ibu Nurul Fajrin selaku dosen pembimbing dan dosen wali yang telah memberikan nasihat, arahan, dan bantuan sehingga penulis dapat menyelesaikan Tugas Akhir ini.
5. Bapak dan Ibu dosen Teknik Informatika ITS yang telah membina dan memberikan ilmu kepada penulis selama menempuh studi di Teknik Informatika ITS.
6. Laboratorium DTK Teknik Informatika ITS yang sudah membantu pengerjaan Tugas Akhri ini.
7. Serta pihak-pihak lain yang namanya tidak dapat penulis sebutkan satu -persatu.

Bagaimanapun juga penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juli 2017

Mohammad Iqbal Ramadhan

## DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK .....	ix
ABSTRACT .....	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI .....	xv
DAFTAR GAMBAR .....	xix
DAFTAR TABEL .....	xxi
DAFTAR KODE SUMBER .....	xxiii
1. BAB I PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Tujuan.....	2
1.3. Rumusan Permasalahan.....	2
1.4. Batasan Permasalahan .....	3
1.5. Metodologi .....	3
1.6. Sistematika Penulisan.....	4
2. BAB II DASAR TEORI.....	7
2.1. Penelitian Terdahulu.....	7
2.2. Pembelajaran Dalam Kelas .....	8
2.3. Perintah Suara Pada Perangkat Bergerak .....	9
2.4. API Pengenalan Suara pada Android .....	10
2.5. Sistem Operasi Android .....	12
2.6. Wi-fi .....	13
2.7. IntelliJ IDEA Community Edition .....	14
2.8. Android Studio .....	15
2.9. Aplikasi Windows .....	15
2.10. Pustaka <i>Runtime</i> dan <i>Robot</i> .....	16
3. BAB III ANALISIS DAN PERANCANGAN SISTEM.....	19
3.1. Analisis Metode Secara Umum .....	19
3.1.1. Analisis Permasalahan.....	19
3.1.2. Deskripsi Umum Sistem.....	20
3.1.3. Spesifikasi Kebutuhan Perangkat Lunak.....	20
3.2. Perancangan Sistem.....	38

3.2.1.	Perancangan Arsitektur.....	38
3.2.2.	Perancangan Antarmuka.....	39
3.2.3.	Perancangan Proses .....	40
3.2.4.	Perancangan Daftar Perintah .....	47
4.	<b>BAB IV IMPLEMENTASI.....</b>	<b>51</b>
4.1.	Lingkungan Implementasi .....	51
4.2.	Implementasi Proses.....	52
4.2.1.	Implementasi Tahap Deteksi Suara .....	52
4.2.2.	Implementasi Lalu Lintas Data Teks ke Server....	53
4.2.3.	Implementasi Membuka Berkas PowerPoint.....	54
4.2.4.	Implementasi Menjalankan Fungsi Play PowerPoint .....	54
4.2.5.	Implementasi Menjalankan Fungsi Next PowerPoint .....	54
4.2.6.	Implementasi Menjalankan Fungsi Previous PowerPoint .....	55
4.2.7.	Implementasi Menutup PowerPoint .....	55
4.2.8.	Implementasi Membuka Notepad.....	55
4.2.9.	Implementasi Mematikan Komputer .....	55
4.3.	Implementasi Antarmuka Perangkat Lunak .....	55
	<b>BAB V PENGUJIAN DAN EVALUASI .....</b>	<b>57</b>
5.1.	Lingkungan Pengujian.....	57
5.2.	Skenario Uji Coba .....	58
5.3.	Skenario Pengujian 1: Membuka Berkas PowerPoint..	58
5.4.	Skenario Pengujian 2: Menjalankan Fungsi Play PowerPoint .....	60
5.5.	Skenario Pengujian 3 : Menjalankan Fungsi Next PowerPoint .....	62
5.6.	Skenario Pengujian 4 : Menjalankan Fungsi Previous PowerPoint .....	63
5.7.	Skenario Pengujian 5 : Menutup PowerPoint.....	65
5.8.	Skenario Pengujian 6: Membuka Notepad .....	66
5.9.	Skenario Pengujian 7: Mematikan Komputer.....	67
5.10.	Pengujian Kebergunaan.....	69
5.11.	Evaluasi Pengujian .....	72



5.11.1.	Evaluasi Pengujian Fungsionalitas .....	73
5.11.2.	Evaluasi Pengujian Kebergunaan .....	73
5.	BAB VI KESIMPULAN DAN SARAN .....	75
6.1.	Kesimpulan.....	75
6.2.	Saran.....	75
	DAFTAR PUSTAKA.....	77
	BIODATA PENULIS.....	89

*{Halaman ini sengaja dikosongkan}*

## DAFTAR GAMBAR

Gambar 2.3.1. (a) Antarmuka siri keluaran Apple, (b) Antarmuka OK Google pada android.....	10
Gambar 2.5.1 Ponsel Cerdas yang Memanfaatkan Sistem Operasi Android.....	12
Gambar 3.1.1 Diagram Kasus Penggunaan Sistem.....	22
Gambar 3.1.2 Diagram Aktivitas Membuka File Aplikasi PowerPoint .....	25
Gambar 3.1.3 Diagram Aktivitas Menjalankan Fungsi Play PowerPoint .....	27
Gambar 3.1.4 Diagram Aktivitas Menjalankan Fungsi Next PowerPoint .....	29
Gambar 3.1.5 Diagram Aktivitas Menjalankan Fungsi Previous PowerPoint .....	31
Gambar 3.1.6 Diagram Aktivitas Menutup PowerPoint .....	33
Gambar 3.1.7 Diagram Aktivitas Membuka Notepad.....	35
Gambar 3.1.8 Diagram Aktivitas Mematikan Komputer .....	37
Gambar 3.2.2 Rancangan Arsitektur Sistem Perintah Suara .....	38
Gambar 3.2.3 Rancangan Antarmuka Halaman Pembuka .....	39
Gambar 3.2.5 Diagram Alir Proses Deteksi Suara .....	41
Gambar 3.2.6 Diagram Alir Proses Pengambilan Kata Perintah Membuka Berkas.....	42
Gambar 3.2.7 Diagram Alir Menjalankan Fungsi Play PowerPoint .....	43
Gambar 3.2.8 Diagram Alir Menjalankan Fungsi Next PowerPoint .....	44
Gambar 3.2.9 Diagram Alir Menjalankan Fungsi Previous PowerPoint .....	45
Gambar 3.2.10 Diagram Alir Menutup PowerPoint.....	46
Gambar 3.2.11 Diagram Alir Proses Membuka Notepad.....	46
Gambar 3.2.12 Diagram Alir Mematikan Komputer .....	47
Gambar 4.2.1 Kode Semu Untuk Inisiasi Objek <i>RecognizerIntent</i> .....	52

Gambar 4.2.2 Kode Semu Pengolahan Suara Menjadi Teks dari <i>Server</i> .....	53
Gambar 4.2.3 Kode Semu Pengiriman dari Perangkat Bergerak ke <i>Server</i> Komputer .....	53
Gambar 4.2.4 Kode Semu Penerimaan Perintah dari Perangkat Bergerak ke <i>Server</i> Komputer .....	54
Gambar 4.3.1 Gambar Implementasi Antarmuka Pengguna <i>smartphone</i> .....	56
Gambar 4.3.2 Gambar Implementasi Pengubahan Suara ke Teks .....	56
Gambar 5.3.1 Hasil Pengujian Skenario 1 Membuka Berkas PowerPoint .....	59
Gambar 5.3.2 Kondisi Akhir Skenario 1 Membuka Aplikasi si PowerPoint .....	59
Gambar 5.4.1 Pengujian Skenario 1 Menjalankan Fungsi “ <i>play</i> ” .....	61
Gambar 5.4.2 Hasil Pengujian Skenario 1 Menjalankan Fungsi “ <i>play</i> ” .....	61
Gambar 5.5.1 Pengujian Skenario 2 Menjalankan Fungsi “ <i>next</i> ” .....	63
Gambar 5.6.1 Pengujian Skenario 3 Menjalankan fungsi “ <i>previous</i> ” .....	64
Gambar 5.7.1 Pengujian Menutup PowerPoint .....	65
Gambar 5.8.1 Hasil Pengujian Skenario 1 Membuka Notepad ...	67
Gambar 5.9.1 Hasil Pengujian Skenario 1 Mematikan Komputer .....	68

## **DAFTAR TABEL**

Tabel 3.1.1 Daftar Kode Kebutuhan Fungsional Aplikasi .....	21
Tabel 3.1.2 Deskripsi Kasus Penggunaan Sistem .....	23
Tabel 3.1.3 Rincian Alur Kasus Penggunaan .....	23
Tabel 3.1.4 Rincian Alur Kasus Penggunaan (2) .....	24
Tabel 3.1.5 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Play PowerPoint .....	26
Tabel 3.1.6 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Play PowerPoint .....	27
Tabel 3.1.7 Rincian Alur Kasus Penggunaan Menjalankan Next PowerPoint .....	28
Tabel 3.1.8 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Next PowerPoint (2) .....	29
Tabel 3.1.9 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Previous .....	30
Tabel 3.1.10 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Previous PowerPoint (2) .....	31
Tabel 3.1.11 Rincian Alur Kasus Penggunaan Menutup PowerPoint .....	32
Tabel 3.1.12 Rincian Alur Kasus Penggunaan Menutup PowerPoint (2) .....	33
Tabel 3.1.13 Rincian Alur Kasus Penggunaan Membuka Notepad .....	34
Tabel 3.1.14 Rincian Alur Kasus Penggunaan Membuka Notepad (2) .....	35
Tabel 3.1.15 Rincian Alur Kasus Penggunaan Mematikan Komputer .....	36
Tabel 3.1.16 Rincian Alur Kasus Penggunaan Mematikan Komputer (2) .....	37
Tabel 3.2.1 Daftar Perintah Navigasi Aplikasi PowerPoint .....	48
Tabel 3.2.2 Daftar Perintah Buka Tutup Aplikasi .....	49
Tabel 3.2.3 Daftar Perintah Pada Aplikasi .....	49
Tabel 3.2.4 Daftar Perintah Pada Aplikasi (2) .....	50
Tabel 3.2.5 Daftar Nama Alias Aplikasi .....	50

Tabel 4.1.1 Lingkungan Implementasi Sistem .....	51
Tabel 4.1.2 Lingkungan Implementasi Sistem (2).....	52
Tabel 5.3.1 Pengujian Membuka Berkas PowerPoint .....	58
Tabel 5.3.2 Pengujian Membuka Berkas PowerPoint (2).....	59
Tabel 5.3.3 Hasil Pemanggilan Perintah Buka Presentasi.....	60
Tabel 5.4.1 Pengujian Menjalankan Fungsi Play PowerPoint.....	60
Tabel 5.4.2 Pengujian Menjalankan Fungsi Play PowerPoint (2) .....	61
Tabel 5.4.3 Hasil Pemanggilan Perintah Mainkan .....	62
Tabel 5.5.1 Pengujian Menjalankan Fungsi Next PowerPoint ....	62
Tabel 5.5.2 Hasil Pemanggilan Perintah Lanjut .....	63
Tabel 5.6.1 Pengujian Menjalankan Fungsi Previous PowerPoint .....	63
Tabel 5.6.2 Pengujian Menjalankan Fungsi Previous PowerPoint (2) .....	64
Tabel 5.6.3 Hasil Pemanggilan Perintah Kembali.....	64
Tabel 5.7.1 Pengujian Menutup PowerPoint .....	65
Tabel 5.7.2 Hasil Pemanggilan Perintah Tutup Presentasi.....	66
Tabel 5.8.1 Pengujian Membuka Notepad .....	66
Tabel 5.8.2 Hasil Pemanggilan Perintah Buka Notepad.....	67
Tabel 5.9.1 Pengujian Mematikan Komputer.....	67
Tabel 5.9.2 Pengujian Mematikan Komputer (2) .....	68
Tabel 5.9.3 Hasil Pemanggilan Perintah Matikan Komputer .....	68
Tabel 5.10.1 Kuesioner Pengujian Kebergunaan .....	69
Tabel 5.10.2 Kuesioner Pengujian Kebergunaan (2).....	70
Tabel 5.10.3 Daftar Penguji Program.....	70
Tabel 5.10.4 Daftar Penguji Program (2) .....	71
Tabel 5.10.5 Hasil Kuesioner .....	71
Tabel 5.10.6 Hasil Kuesioner (2).....	72
Tabel 5.11.1 Rangkuman Hasil Pengujian .....	73

## **DAFTAR KODE SUMBER**

Kode Sumber 1 activity.xml.....	79
Kode Sumber 2 Colors.xml .....	79
Kode Sumber 3 Kode Sumber Mengirim Pesan Perintah .....	81
Kode Sumber 4 Kode Sumber Pengecekan Kode .....	82
Kode Sumber 5 Kode Sumber Deteksi Suara.....	83
Kode Sumber 6 Kode Sumber Menutup Program.....	84
Kode Sumber 7 Kode Sumber Menerima Koneksi Perangkat Bergerak .....	84
Kode Sumber 8 Kode Sumber Membuka PowerPoint .....	85
Kode Sumber 9 Kode Sumber Menjalankan Fungsi PowerPoint	86
Kode Sumber 10 Kode Sumber Mematikan Komputer.....	87
Kode Sumber 11 Kode Sumber Menutup Program.....	87

*[Halaman ini sengaja dikosongkan]*



# **BAB I**

## **PENDAHULUAN**

Pada bab ini akan dipaparkan mengenai garis besar Tugas Akhir yang meliputi latar belakang, tujuan, rumusan dan batasan permasalahan, metodologi pembuatan Tugas Akhir, dan sistematika penulisan.

### **1.1. Latar Belakang**

Pengembangan aplikasi pengendali jarak jauh sangat dibutuhkan saat ini. Penggunaan perangkat yang dalam operasionalnya tidak boleh menggunakan tangan membuat aplikasi pengendali jarak jauh menjadi keunggulan tersendiri. Dengan berbagi jaringan yang sama, dua perangkat atau lebih, dapat dikendalikan oleh pengguna untuk suatu kebutuhan tertentu. Pengendalian ini bentuknya bermacam macam, salah satunya adalah pengendalian perangkat dengan suara.

Pengenalan suara-ke-teks adalah proses menerjemahkan gelombang suara hasil pengucapan kata-kata menjadi bentuk teks yang sesuai. Saat ini, pengenalan suara-ke-teks telah banyak diterapkan pada perangkat ponsel cerdas. Pengenalan suara-ke-teks banyak diterapkan untuk menulis keluaran suara menjadi teks atau menjalankan perintah tertentu tanpa harus mengetikkan sendiri isi teks atau menekan tombol yang bersesuaian. Dengan menggunakan teknologi ini, manusia dengan ponsel cerdas miliknya, bisa memberi perintah suara yang diubah menjadi teks terhadap perangkat lain yang tersambung dalam suatu jaringan. Atas dasar itulah, muncul ide untuk menggabungkan keduanya dengan membangun aplikasi pengenalan suara untuk mengoperasikan komputer berbasis Android menggunakan Bahasa Indonesia.

Contoh penggunaan pengendalian jarak jauh adalah pengoperasian PowerPoint, membuka program yang ada pada sistem operasi komputer, dan mematikan komputer.

Pengoperasian PowerPoint menjadi fundamental bila pengguna ingin mengoperasikan PowerPoint tanpa menggunakan tombol sehingga pengguna terbantu dalam menyiapkan presentasi.

Berdasarkan latar belakang di atas, Teknologi pengenalan suara-ke-teks yang digunakan pada aplikasi ini menggunakan API masukan yang disediakan oleh Google. Respon yang diberikan aplikasi pada pengguna yaitu berupa berjalannya aplikasi powerpoint pada komputer sesuai dengan perintah yang diucapkan oleh pengguna. Perintah suara yang diberikan adalah mainkan, lanjut, dan kembali.

## **1.2. Tujuan**

Tujuan dari pembuatan Tugas Akhir ini adalah menerapkan pengenalan suara-ke-teks berbahasa Indonesia pada ponsel cerdas berbasis Android, membuat suatu aplikasi yang dapat mengoperasikan aplikasi PowerPoint dengan menggunakan perintah suara berbasis android dengan memanfaatkan ponsel cerdas, dan mengatur interaksi antara pengguna dengan aplikasi melalui perintah suara.

## **1.3. Rumusan Permasalahan**

Rumusan masalah yang diangkat dalam tugas akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana menerapkan pengenalan suara-ke-teks Bahasa Indonesia pada ponsel cerdas berbasis Android?
2. Bagaimana merancang aplikasi pendeteksi perintah suara berbahasa Indonesia untuk mengoperasikan komputer menggunakan ponsel cerdas berbasis android?
3. Bagaimana mengatur interaksi antara pengguna dengan aplikasi melalui perintah suara?

#### 1.4. Batasan Permasalahan

Beberapa batasan masalah yang menjadi batas pada tugas akhir ini adalah:

1. Alat pendeteksi suara-ke-teks menggunakan ponsel cerdas berbasis android.
2. Bahasa pemrograman *server* menggunakan java.
3. Perintah suara menggunakan Bahasa Indonesia.
4. *Server* tidak menggunakan GUI.

#### 1.5. Metodologi

Adapun beberapa tahap dalam proses pengerjaan tugas akhir ini, yaitu sebagai berikut:

##### 1. Studi Literatur

Pada studi literatur ini, akan dipelajari sejumlah referensi yang diperlukan dalam pembuatan aplikasi. Adapun tahap studi literature dan pemahaman system adalah sebagai berikut.

- Mempelajari beberapa kakas atau pustaka untuk pengenalan suara-ke-teks dan teks-ke-suara
- Mempelajari tentang penggunaan *Google Voice Recognition* yang ada pada ponsel cerdas.
- Mempelajari literatur mengenai pengembangan aplikasi pada platform android.
- Mempelajar konsep *client-server* pada android dan komputer, android studio, dan API yang terkait.

##### 2. Implementasi

Implementasi yang dilakukan adalah merancang bangun *voice recognition* pada ponsel cerdas untuk mengoperasikan komputer dengan konsep *client-server*.

Implementasi *server* ini dibangun dengan menggunakan Bahasa pemrograman Java dengan menggunakan IDE IntelliJ IDEA Community dan pengembangan aplikasi android menggunakan IDE Android Studio.

### **3. Uji coba dan evaluasi**

Pada tahap ini, akan dilakukan uji coba mengoperasikan PowerPoint menggunakan perintah suara berbasis android menggunakan ponsel cerdas. Beberapa hal yang akan diujicobakan adalah sebagai berikut.

1. Membuka berkas PowerPoint.
2. Menjalankan fungsi play PowerPoint.
3. Menjalankan fungsi next Powerpoint.
4. Menjalankan fungsi previous PowerPoint.
5. Menutup PowerPoint.
6. Membuka ntotepad.
7. Mematikan komputer.

### **4. Penyusunan buku tugas akhir**

Tahap ini merupakan tahap penyusunan laporan berupa buku tugas akhir sebagai dokumentasi pelaksanaan tugas akhir, yang mencakup seluruh teori, implementasi, serta hasil pengujian yang telah dikerjakan. Kesimpulan dan saran diberikan pada buku Tugas akhir ini sebagai tambahan terhadap aplikasi.

## **1.6. Sistematika Penulisan**

Buku Tugas Akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan

pengembangan lebih lanjut. Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini.

### **Bab I Pendahuluan**

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan Tugas Akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan Tugas Akhir.

### **Bab II Dasar Teori**

Memaparkan hasil studi literature yang digunakan menyelesaikan Tugas Akhir ini, terdiri atas, pustaka pengenalan suara-ke-teks dan pengenalan teks-ke-suara, *Google Voice Recognition API*, Android SDK, dan penerapan *client-server* pada pengembangan aplikasi Tugas Akhir ini.

### **Bab III Analisis dan Perancangan Sistem**

Membahas rancangan system yang dibangun dengan pemodelan UML, dimulai dari deskripsi umum sistem, batasan sistem, kebutuhan fungsional, imteraksi system dengan pengguna dan rancangan antarmuka pengguna.

### **Bab IV Implementasi**

Bab ini berisi implementasi dari perancangan dan implementasi fitur-fitur penunjang aplikasi termasuk implementasi sistem dan implementasi antarmuka pengguna.

### **Bab V Pengujian dan Evaluasi**

Bab ini membahas pengujian dengan metode kotak hitam (*black box testing*) untuk mengetahui aspek nilai fungsionalitas dari perangkat lunak dan nilai kegunaan yang dibuat dengan juga memperhatikan ketertarikan pada calon partisipan untuk menggunakan aplikasi ini.

### **Bab VI Kesimpulan dan Saran**

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan pada Tugas Akhir ini. Bab ini

membahas saran-saran untuk pengembangan sistem lebih lanjut.

**Daftar Pustaka**

Merupakan daftar referensi yang digunakan untuk mengembangkan Tugas Akhir.

**Lampiran**

Merupakan bab tambahan yang berisi daftar kode yang ada pada aplikasi ini.

## **BAB II**

### **DASAR TEORI**

Pada Bab ini akan dibahas mengenai teori-teori yang menjadi dasar dari pembuatan Tugas Akhir. Teori-teori tersebut meliputi pengertian dan beberapa analisis terkait pengoperasian PowerPoint menggunakan android.

#### **2.1. Penelitian Terdahulu**

Penelitian terdahulu yang relevan adalah penelitian yang dilakukan oleh Fajar Maulana Putra <sup>[1]</sup>, alumni Teknik Informatika angkatan 2009, yang berjudul “Rancang Bangun Perangkat Lunak Pengirim SMS Dengan Perintah Suara Berbahasa Indonesia Pada Platform Android”. Perangkat lunak yang dibuat adalah perangkat bergerak menggunakan ponsel cerdas berbasis android untuk mengirim sms dengan menggunakan suara dan memanfaatkan Dragon SDK untuk pengenalan suara dan pembangkit teks-ke-suara serta Android SDK dalam pengembangannya.

Suara masukan diterjemahkan dengan menggunakan algoritma *stemming* gagasan Nazief dan Andriani. Hasil dari algoritma *stemming* kemudian digunakan untuk menentukan perintah SMS yang diminta oleh pengguna. Proses pengiriman dan penerimaan sms menggunakan Android SMS API.

Dalam hasil penelitian tersebut, menunjukkan bahwa aktifitas atau perintah pada perangkat lunak pengirim sms menggunakan suara ini telah berhasil seperti mengirimkan pesan, menerima pesan, serta menghapus pesan. Dragon SDK dalam hal ini digunakan untuk pengenalan suara dan pembangkitan teks-ke-suara.

Penelitian yang lain adalah penelitian yang dilakukan oleh Muhammad Najib <sup>[2]</sup>, alumni Teknik Informatika 2009, yang berjudul “Rancang Bangun Aplikasi Penuntun Jalan Dengan Perintah Suara Berbahasa Indonesia Pada Perangkat Bergerak Berbasis Android”.

Dalam penelitian tersebut, peneliti menggunakan Android SDK sebagai kakas bantu pengembangan dan Memanfaatkan Google Maps API. Sebagai kakas bantu pengenalan suaranya digunakan Google Voice Recognition API yang sudah disediakan bersama dengan Google Maps API.

Suara masukan diterjemahkan menjadi petunjuk jalan pada perangkat bergerak sehingga memungkinkan pengguna untuk mengetahui arah dan jalur yang akan dilewati untuk sampai ke tujuan. Dengan adanya Google Maps API maka system bisa dengan akurat mengetahui lokasi pengguna yang menggunakan perangkat lunak hasil penelitian tersebut.

Hasil penelitian tersebut menunjukkan bahwa perintah suara pada perangkat bergerak untuk penunjuk jalan telah membuahkan hasil terbukti dari keluarnya hasil arah jalan yang akan dilalui oleh pengguna perangkat lunak tersebut.

## **2.2. Pembelajaran Dalam Kelas**

Penyampaian mata pelajaran yang kurang menarik dengan cara mengajar bisa digantikan dengan pembelajaran melalui Powerpoint. Menurut Elmi Mahzum (2008:2), Microsoft PowerPoint adalah media yang paling banyak digunakan sebagai media presentasi baik oleh korporat maupun oleh kalangan biasa<sup>[3]</sup>. Penyampaian dengan PowerPoint tidak seperti mengajar dimana pengajar harus menerangkan sekaligus menulis, namun lebih kepada pengajar untuk mempersiapkan dahulu apa poin yang ingin disampaikan kepada anak didik barulah pengajar menjelaskan poin-poin yang terdapat pada PowerPoint yang dimilikinya.

Beberapa kegunaan PowerPoint dalam media pembelajaran adalah sebagai berikut<sup>[4]</sup>:

1. Memperjelas penyajian pesan agar tidak terlalu bersifat verbalistis.
2. Mengatasi keterbatasan ruang, waktu, dan daya indera.



3. Penggunaan media pembelajaran secara tepat dan bervariasi dapat mengatasi sikap pasif anak didik.
4. Sifat yang unik pada tiap siswa dalam menerima pengajaran dapat disatukan dengan media pembelajaran PowerPoint.

Karena PowerPoint paling banyak digunakan dalam media pembelajaran maka perkembangan selanjutnya dalam pembelajaran di dalam kelas juga ikut bertambah seperti adanya layar LCD sehingga PowerPoint bisa dilihat tidak hanya oleh pengajar namun anak didik juga. Adanya penunjuk yang menggunakan laser sehingga memudahkan pengajar untuk menunjuk poin yang ingin disampaikan kepada anak didik.

Perkembangan pengajaran menggunakan PowerPoint juga dikembangkan pada perangkat bergerak berbasis android. Dapat diambil contoh adanya perangkat *mouse* tiruan sehingga pengajar tidak perlu dekat atau tergantung dengan perangkat *mouse* melainkan dengan perangkat bergerak berbasis android miliknya yang sudah ditanam aplikasi perangkat *mouse* tiruan. Layaknya sebuah perangkat *mouse*, pengguna dapat menggerakkan *cursor* pada layar komputer dan melakukan perintah *klik* kanan dan kiri. Dengan memanfaatkan teknologi pada perangkat bergerak berbasis android, pengembang dapat memodifikasi bagaimana dalam menggantikan perangkat *mouse* menjadi teknologi suara sehingga perintah suara pada perangkat bergerak berbasis android yang diinginkan akan dilaksanakan oleh komputer untuk menjalankan PowerPoint.

### 2.3. Perintah Suara Pada Perangkat Bergerak

Perintah suara pada perangkat bergerak merupakan sebuah perintah yang diucapkan dari suara manusia untuk menjalankan aktivitas tertentu pada perangkat bergerak. Saat ini banyak perangkat bergerak yang sudah memiliki fitur perintah suara.



Siri dan Ok Google merupakan aplikasi yang

**Gambar 2.3.1. (a) Antarmuka siri keluaran Apple, (b) Antarmuka OK Google pada android**

Google biasanya terdapat pada perangkat bergerak berbasis Android.

Pada gambar 2.2.1 bagian a merupakan aplikasi pengenalan suara yang dikembangkan oleh Apple yaitu siri. Siri dapat melakukan pengiriman pesan teks, panggilan telepon, pengingat, dan lain-lain. Sedangkan pada bagian b merupakan aplikasi OK Google yang memanfaatkan suara untuk membantu melakukan pencarian pada pengaya Google Chrome. Aplikasi ini bisa melakukan pencarian layaknya *search engine* google versi mini untuk mencari apa saja yang diinginkan oleh pengguna.

## 2.4. API Pengenalan Suara pada Android

API masukan suara pada android berupa sebuah layanan yang disediakan oleh Google sebagai kakas bantu aplikasi dalam pengenalan suara yang terdengar di sekitar ponsel cerdas dan diterjemahkan dalam bentuk teks. Layanan ini sudah tersedia pada perangkat bergerak berbasis android. Sifat dari android yang *open platform* inilah yang membuat android dapat dikembangkan dengan mudah dan gratis <sup>[5]</sup>.

Penggunaan API pengenalan suara oleh Google ini dilakukan dengan mengalirkan gelombang suara ke *server* Google untuk kemudian dicari dan dicocokkan. Dari hasil pengenalan dan pencocokan yang dilakukan *server* maka akan dikirimkan kembali pada *client* dalam hal ini pengguna perangkat bergerak, kata yang cocok dengan suara yang telah dikenali tersebut.

API ini mendukung banyak sekali Bahasa diantaranya adalah Bahasa Indonesia. Dengan menggunakan Android Studio,

pengembang dapat dengan mudah menggunakan kakas bantu pengenalan suara ini dimulai dengan penerimaan *RecognizerIntent* pada *client*. Kemudian setelah *RecognizerIntent* diterima, perangkat bergerak akan merekam suara yang ada disekitar perangkat tersebut kemudian melakukan pengiriman data ke *server* Google untuk melakukan pencocokan basis data teks dengan suara yang telah dikirim. Untuk melakukan hal ini pengguna wajib menyambungkan ponsel cerdasnya ke wifi dengan atau tanpa proxy. Selain itu, sambungan ini juga dibutuhkan oleh ponsel cerdas dan *server*, dalam hal ini komputer, untuk bisa berkomunikasi dalam pengiriman dan penerimaan perintah yang diberikan oleh pengguna ke komputer. Sama seperti pustaka pengenalan suara pada umumnya, pengenalan suara pada API Suara Google juga dipengaruhi oleh beberapa hal, diantaranya adalah <sup>[6]</sup>:

- Cara pengucapan. Pengucapan yang lantang akan menghasilkan keluaran yang lebih akurat.
- Tempo pengucapan. Tempo pengucapan yang lambat atau normal dapat memberikan keluaran yang lebih akurat dibandingkan dengan pengucapan yang cepat.
- Penggunaan kata “eh”, “ehm”, dan sejenisnya juga akan mempengaruhi keluaran suara.
- Lingkungan sekitar. Lingkungan yang memiliki sedikit *noise* atau sepi akan memberikan keluaran yang lebih bagus daripada lingkungan yang ramai.

Beberapa kelebihan yang ditawarkan oleh API Suara Google diantaranya:

- Mendukung lebih dari 20 bahasa, termasuk Bahasa Indonesia.
- Tidak sulit dalam penggunaannya.
- Pengenalan suara mudah dikembangkan sesuai dengan keinginan pengembang.
- Keluaran yang dihasilkan cukup akurat.
- Bisa dalam bentuk *online* dan *offline*.

## 2.5. Sistem Operasi Android

Sistem yang dibangun dalam tugas akhir ini adalah berupa aplikasi perangkat bergerak yang menjalankan system operasi Android. Android merupakan system operasi berbasis *Linux* yang digunakan pada gadget yang dikembangkan oleh Google. Android merupakan *open platform* dan tersedia bagi pengguna yang ingin mengembangkan perangkat lunak menggunakan system operasi Android<sup>[7]</sup>.

Android memiliki banyak layanan diantaranya Gmail, Google Calendar, Google Maps, dan mesin pencari Google karena pengembangnya adalah Google. Android tidak terbatas hanya layanan google saja yang ada namun ada beberapa layanan yang lain seperti pengiriman pesan, alarm, pengingat, *widget*, dll. Namun yang pasti ada pada system operasi android *default-nya* adalah aplikasi layanan Google<sup>[8]</sup>.



**Gambar 2.5.1 Ponsel Cerdas yang Memanfaatkan Sistem Operasi Android**

Pada gambar 2.4.1 terlihat banyak sekali merk dagang perangkat bergerak yang menggunakan system operasi Android. Dalam perkembangannya sistem operasi android mengalami peningkatan dari beberapa versi. Adapun versi system operasi android adalah sebagai berikut:

1. Android versi 1.1
2. Android versi 1.5 (Cupcake)
3. Android versi 1.6 (Donut)
4. Android versi 2.0 (Éclair)

5. Android versi 2.1 (Éclair)
6. Android versi 2.2 (Froyo : Frozen Yoghurt)
7. Android versi 2.3 (Gingerbread)
8. Android versi 3.0/3.1 (Honeycomb)
9. Android versi 4.0 (ICS : Ice Cream Sandwich)
10. Android versi 4.1 (Jelly Bean)
11. Android versi 4.4 (Kitkat)
12. Android versi 5.0 (Lollipop)
13. Android versi 6.0 (Marshmallow)
14. Android versi 7.0 (Nougat)

## 2.6. Wi-fi

*Wi-fi* atau yang biasa kita sebut “*waifai*” merupakan sebuah teknologi komunikasi antar perangkat yang terhubung dalam suatu jaringan nirkabel. Artinya penggunaan *wi-fi* sama dengan penggunaan LAN yaitu supaya perangkat yang terhubung pada jaringan *wi-fi* tersebut bisa melakukan koneksi internet secara *online*. Untuk dapat melakukan koneksi internet, pengguna menghubungkan perangkat miliknya yaitu ponsel cerdas ke jaringan *wi-fi* yang tersedia.

Dalam perkembangannya, *wi-fi* dirancang berdasarkan spesifikasi IEEE 802.11<sup>[9]</sup>. Sekarang ada 4 variasi dari 802.11, yaitu:

- 802.11a
- 802.11b
- 802.11g
- 802.11n

Yang membedakan antara keempat variasi tersebut adalah kecepatan dan kecocokannya.

*Wi-fi* juga memiliki banyak fungsi sebagai berikut <sup>[10]</sup>:

### 1. Sebagai pengaman.

Aplikasi seperti “find my iPhone”, menggunakan *wi-fi* untuk mencari iPhone yang lupa meletakkannya. Untuk kita yang tidak menggunakan iPhone bisa menggunakan program “Prey” dimana kita bisa melacak HP melalui komputer atau

laptop saat HP kita dicuri dan apa saja yang dilakukan pencuri kepada HP kita.

2. Berbagi berkas dengan komputer atau perangkat lain.  
Penyedia berkas penyimpanan daring seperti Dropbox memungkinkan kita bisa berbagi foto, dokumen, video, dan lainnya dengan sangat mudah.
3. Berguna sebagai router nirkabel.  
Sebuah aplikasi yang bernama PdaNet yang dapat mengubah android ke router wi-fi yang sangat berguna sekali untuk PC.
4. Untuk melihat film di TV secara daring.  
Kegunaan wi-fi yaitu sebagai media hiburan menonton film secara daring sehingga pengguna tidak merasa bosan.
5. Sinkronisasi HP tanpa USB  
Dengan menggunakan aplikasi berbagi dokumen dari HP ke komputer dapat digunakan wi-fi sebagai medianya.

## 2.7. IntelliJ IDEA Community Edition

Merupakan kakas pengembang terpadu dari JetBrains yang dikembangkan oleh IntelliJ. Selain itu IntelliJ juga mengembangkan IDE Android Studio. IntelliJ IDEA merupakan IDE (Integrated Development Environment) yang dapat dapat dijalankan di berbagai platform, seperti Windows, OS X, dan Linux. Pengoperasian IntelliJ mudah karena tidak perlu melakukan pengaturan JRE sehingga tidak perlu instalasi Java terlebih dahulu untuk membuat aplikasi yang dibuat pada IntelliJ IDEA.

IntelliJ IDEA Community Edition mendukung koneksi dengan GitHub sehingga dapat melakukan *push* dan *pull* seperti pada aplikasi GitHub. IntelliJ IDEA berbayar mendukung banyak Bahasa pemrograman, namun yang tidak berbayar sudah mendukung Bahasa pemrograman Java. Aplikasi IDE IntelliJ IDEA dapat diunduh secara *online* pada website JetBrains. IDE ini dibagikan secara gratis untuk Community Edition sedangkan Ultimate Edition adalah versi yang berbayar.

## 2.8. Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu – Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA . Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya:

- Sistem pembuatan berbasis Gradle yang fleksibel
- Emulator yang cepat dan kaya fitur
- Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
- Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
- Template kode dan koneksi ke GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
- Alat penguji dan kerangka kerja yang ekstensif
- Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain
- Dukungan C++ dan NDK
- Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.

## 2.9. Aplikasi Windows

Merupakan program yang berjalan pada Sistem Operasi Windows. Sistem Operasi Windows terbagi menjadi Sistem Operasi Windows 32-bit dan 64-bit. Sistem Operasi Windows 64-bit bisa menjalankan aplikasi

windows 64-bit dan 32-bit sedangkan Sistem Operasi Windows 32-bit hanya bisa menjalankan aplikasi 32-bit saja.

Bentuk pengoperasian komputer bermacam-macam. Beberapa bentuknya yaitu:

1. Pengoperasian Protokol Aplikasi  
Pengoperasian protokol aplikasi merupakan bentuk pengoperasian oleh pengguna terhadap cara jalan aplikasi. Misal dalam PowerPoint merupakan protokol menjalankan fungsi *play*, *next*, *previous*.
2. Buka Tutup Aplikasi  
Pengguna komputer dapat melakukan operasi buka tutup aplikasi. Sehingga jika pengguna ingin membuka aplikasi, pengguna tinggal mencari aplikasi yang diinginkan begitu juga bila pengguna ingin menutup aplikasi, pengguna tinggal mencari aplikasi mana yang ingin ditutup.
3. Pengoperasian Sistem Komputer  
Pengoperasian sistem komputer banyak macamnya. Salah satunya adalah mematikan dan menghidupkan ulang pada komputer. Biasanya setelah pemakaian oleh pengguna, maka komputer akan dimatikan dengan perintah `alt + f4` dalam *shortcut keyboard* atau bisa menekan `windows + shutdown`. Sedangkan bila ingin menghidupkan ulang komputer tanpa mematikannya, pengguna tinggal menjalankan proses *restart* komputer.

## 2.10. Pustaka *Runtime* dan *Robot*

Pustaka *runtime* dapat dipanggil pada *class* `java.lang.Runtime`<sup>[11]</sup>. Kelas ini memungkinkan aplikasi berinteraksi dengan lingkungan tempat aplikasi dijalankan. Kelas ini tidak perlu diimport karena tergabung pada kelas `java.lang.*`. Kelas *runtime* merupakan kelas abstrak yang menggunakan *private Constructor* sehingga tidak dapat dipanggil kelas instannya di luar kelas. Contoh pemanggilan kelas *runtime* adalah `Runtime run = Runtime.getRuntime();`. Setiap aplikasi yang ada memiliki objek



runtime yang dapat diakses menggunakan `getRuntime()`. Sehingga aplikasi dapat berinteraksi dengan lingkungan yang ada di sekitarnya.

Pustaka *Robot* dapat dipanggil pada *class* `java.awt.robot`. Kelas ini merupakan kelas abstrak. Contoh pemanggilan kelas *robot* adalah `Robot robot = new Robot()`. Kelas ini dapat menggantikan pemanggilan *keyboard* dan *mouse* sehingga pengoperasian komputer yang melalui perangkat *keyboard* dan *mouse* bisa digantikan menggunakan kelas ini.

*{Halaman ini sengaja dikosongkan}*

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Pada Bab 3 ini akan dijelaskan mengenai analisis dan perancangan perangkat lunak untuk mencapai tujuan dari tugas akhir. Perancangan ini meliputi perancangan data, perancangan proses, dan perancangan antar muka, serta juga akan dijelaskan tentang analisis implementasi metode secara umum pada sistem.

#### **3.1. Analisis Metode Secara Umum**

Pada subbab berikut akan dijelaskan analisis perangkat lunak untuk pengoperasian komputer menggunakan Bahasa Indonesia. Analisis yang dilakukan meliputi analisis permasalahan, kebutuhan umum perangkat lunak, deskripsi umum sistem, arsitektur dan kebutuhan fungsional sistem.

##### **3.1.1. Analisis Permasalahan**

Permasalahan utama yang diangkat pada pembuatan Tugas Akhir ini adalah bagaimana menerapkan perintah suara pada suatu aplikasi pada ponsel cerdas berbasis android serta mengatur interaksi antara pengguna dengan aplikasi pada komputer.

Untuk mempermudah sistem dalam melakukan perintah pengenalan suara, maka perintah suara sudah berada pada bentuk dasarnya. Sehingga susunan perintah yang diberikan adalah kata dasar dari perintah ditambah dengan kata berikutnya. Dengan demikian, pengguna dapat mengoperasikan aplikasi PowerPoint pada komputer dengan perintah tersebut.

Kemampuan pengubahan dari suara ke teks yang digunakan dalam Tugas Akhir ini adalah dengan memanfaatkan API Suara Google. Penggunaan API Suara Google dikarenakan sudah mendukung Bahasa Indonesia serta sudah ada pada ponsel cerdas berbasis android.

Untuk menghubungkan antara ponsel cerdas dan komputer, digunakan jaringan wi-fi. Jaringan wi-fi nantinya digunakan sebagai jembatan untuk jalan komunikasi antara ponsel cerdas dengan komputer. Jaringan wi-fi yang dibutuhkan beragam, bisa dengan jaringan yang menggunakan proxy ataupun tanpa proxy, yang terpenting adalah port yang digunakan *server* dan ponsel cerdas harus sama.

### **3.1.2. Deskripsi Umum Sistem**

Pada Tugas Akhir ini akan dibuat sebuah perangkat lunak berupa aplikasi berbasis Sistem Operasi Android yang memanfaatkan teknologi input suara untuk mengoperasikan PowerPoint. Suara yang diinputkan adalah suara berbahasa Indonesia yang digunakan langsung pada saat aplikasi ini dijalankan. Perangkat lunak dihubungkan dengan *server* yang ada pada komputer dengan menggunakan jaringan wi-fi yang sudah tersedia.

Respon yang dihasilkan dari aplikasi ini setelah menerima permintaan oleh pengguna adalah berupa berjalannya perintah-perintah yang disebutkan pengguna di aplikasi PowerPoint. Pengenalan suara pengguna dilakukan oleh perangkat Android dengan memanfaatkan API masukan suara pada Android.

### **3.1.3. Spesifikasi Kebutuhan Perangkat Lunak**

Subbab ini membahas spesifikasi kebutuhan perangkat lunak dari hasil analisis yang telah dilakukan. Bagian ini berisi kebutuhan perangkat lunak yang direpresentasikan dalam bentuk kebutuhan fungsional, diagram kasus penggunaan, dan diagram aktivitas.

#### **3.1.3.1. Kebutuhan Fungsional**

Kebutuhan fungsional mendefinisikan layanan yang harus dimiliki oleh perangkat lunak, reaksi dari perangkat lunak terhadap suatu masukan, hasil yang dilakukan perangkat lunak pada situasi

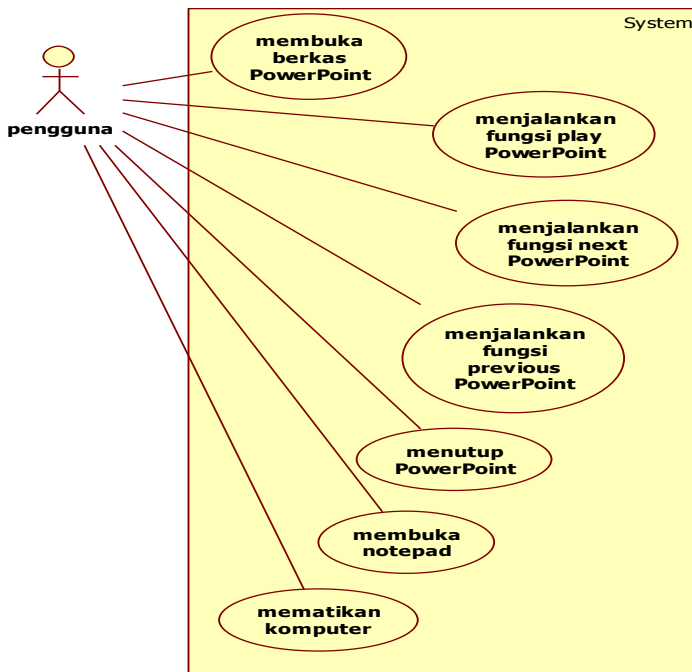
khusus. Kebutuhan fungsional dari perangkat lunak untuk mengoperasikan komputer menggunakan suara dijelaskan pada Tabel 3.1.3.1

**Tabel 3.1.1 Daftar Kode Kebutuhan Fungsional Aplikasi**

<b>Kode Kebutuhan Fungsional</b>	<b>Kebutuhan Fungsional</b>	<b>Deskripsi</b>
FR01	Membuka File Aplikasi PowerPoint	Melakukan buka file PowerPoint pada folder yang telah ada
FR02	Menjalankan Fungsi Play PowerPoint	Menjalankan perintah play pada PowerPoint
FR03	Menjalankan Fungsi Next Powerpoint	Menjalankan perintah next pada PowerPoint
FR04	Menjalankan Fungsi Previous PowerPoint	Menjalankan perintah previous pada PowerPoint
FR05	Menutup PowerPoint	Menutup program PowerPoint yang telah dibuka
FR06	Mematikan Komputer	Mematikan komputer tanpa melalui tombol melainkan dengan suara.
FR07	Membuka Notepad	Membuka aplikasi notepad

### 3.1.3.2. Diagram Kasus Penggunaan

Kasus penggunaan perangkat lunak dijelaskan secara rinci dalam subbab ini. Kasus penggunaan didasarkan pada hasil analisis kebutuhan fungsional dari perangkat lunak. Terdapat tujuh kasus penggunaan pada perangkat lunak ini, sebagaimana ditunjukkan pada Gambar 3.1, yaitu kasus penggunaan untuk membuka berkas PowerPoint, menjalankan fungsi play PowerPoint, menjalankan fungsi next PowerPoint, menjalankan fungsi previous PowerPoint, menutup PowerPoint, membuka notepad, dan mematikan komputer.



Gambar 3.1.1 Diagram Kasus Penggunaan Sistem

Penjelasan lengkap mengenai kasus penggunaan dapat dilihat pada tabel 3.1.2.

**Tabel 3.1.2 Deskripsi Kasus Penggunaan Sistem**

<b>Kode Kasus Penggunaan</b>	<b>Nama Kasus Penggunaan</b>
UC01	Membuka File PowerPoint
UC02	Menjalankan Fungsi Play PowerPoint
UC03	Menjalankan Fungsi Next PowerPoint
UC04	Menjalankan Fungsi Previous PowerPoint
UC05	Menutup PowerPoint
UC06	Membuka Notepad
UC07	Mematikan Komputer

### **3.1.3.3. Kasus Penggunaan Membuka File PowerPoint**

Pada kasus penggunaan membuka program, pengguna menekan tombol suara untuk memasukkan perintah membuka berkas PowerPoint. Sistem akan membuka berkas PowerPoint yang telah disebutkan.

Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.3.

**Tabel 3.1.3 Rincian Alur Kasus Penggunaan**

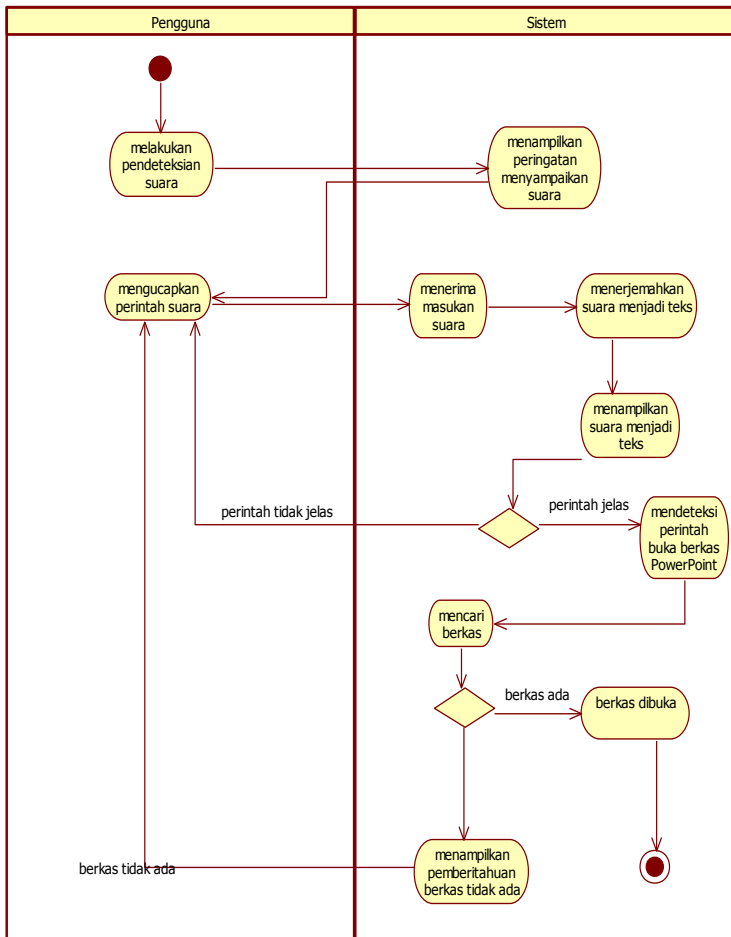
<b>Nama Kasus Penggunaan</b>	Membuka File Aplikasi PowerPoint
<b>Nomor</b>	UC01
<b>Aktor</b>	Pengguna

**Tabel 3.1.4 Rincian Alur Kasus Penggunaan (2)**

<b>Kondisi Awal</b>	Perintah suara telah dikirim ke <i>server</i>
<b>Kondisi Akhir</b>	Komputer membuka berkas PowerPoint
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna mengucapkan perintah dalam Bahasa Indonesia</li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li> <li>7. Sistem mendeteksi perintah membuka berkas PowerPoint</li> <li>8. Sistem mencari berkas yang dimaksudkan A2. Berkas tidak ada</li> <li>9. Sistem membuka berkas PowerPoint</li> </ol>
<b>Alur Alternatif</b>	<p>A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik A.1.1 Kembali ke alur normal nomor 3</p> <p>A2. Berkas tidak ada A.2.1 Sistem memberi pemberitahuan jika berkas tidak ada A.2.2 Kembali ke alur normal nomor 3</p>



Berdasarkan skenario kasus penggunaan pada Tabel 3.1.3. Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan membuka program dapat dilihat pada Gambar 3.1.2.



**Gambar 3.1.2 Diagram Aktivitas Membuka File Aplikasi PowerPoint**

### 3.1.3.4. Kasus Penggunaan Menjalankan Fungsi play PowerPoint

Pada kasus penggunaan menjalankan fungsi play PowerPoint, pengguna dapat melakukan fungsi pada program melalui perintah suara Bahasa Indonesia berupa *play* pada Powerpoint sehingga PowerPoint berada dalam tahap *fullscreen*. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.5.

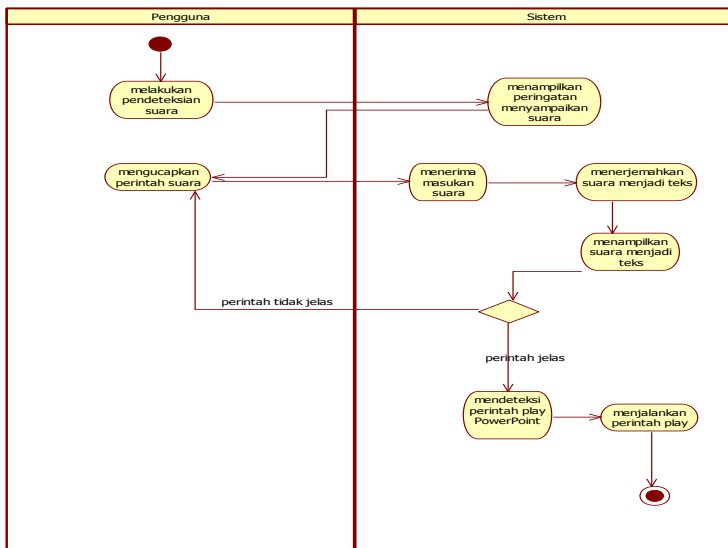
**Tabel 3.1.5 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Play PowerPoint**

<b>Nama Kasus Penggunaan</b>	Menjalankan Fungsi Play PowerPoint
<b>Nomor</b>	UC02
<b>Aktor</b>	Pengguna
<b>Kondisi Awal</b>	Program PowerPoint sudah terbuka
<b>Kondisi Akhir</b>	Sistem menjalankan fungsi play pada PowerPoint
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia</li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li> </ol>

**Tabel 3.1.6 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Play PowerPoint**

<b>Alur Normal</b>	<p>7. Sistem mendeteksi perintah menjalankan fungsi play pada Powerpoint</p> <p>8. Sistem menjalankan fungsi play pada PowerPoint</p>
<b>Alur Alternatif</b>	<p>A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</p> <p>A1.1. Kembali ke alur normal nomor 3</p>

Berdasarkan skenario kasus penggunaan pada Tabel 3.1.4 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menjalankan fungsi play PowerPoint dapat dilihat pada Gambar 3.1.3.



**Gambar 3.1.3 Diagram Aktivitas Menjalankan Fungsi Play PowerPoint**

**3.1.3.5. Kasus Penggunaan Menjalankan Fungsi Next PowerPoint**

Pada kasus penggunaan menjalankan fungsi next PowerPoint, pengguna dapat melakukan fungsi pada program melalui perintah suara Bahasa Indonesia berupa *next* pada Powerpoint sehingga PowerPoint dapat memindahkan ke *slide* selanjutnya. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.7.

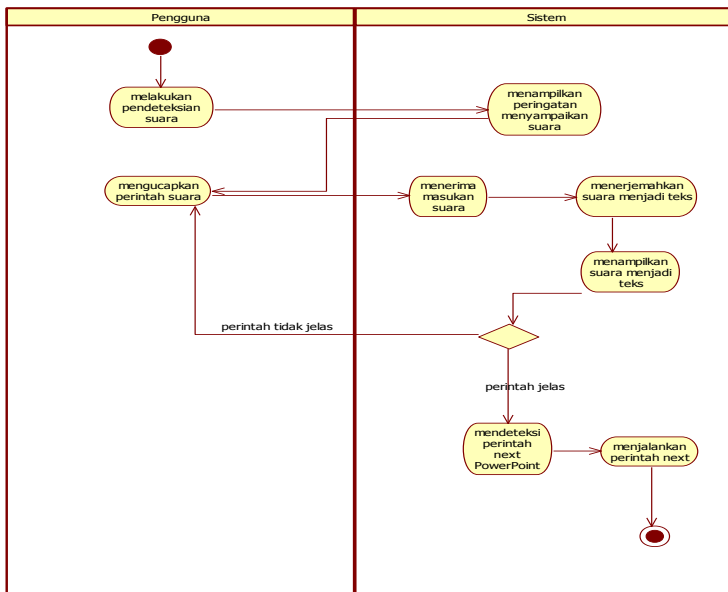
**Tabel 3.1.7 Rincian Alur Kasus Penggunaan Menjalankan Next PowerPoint**

<b>Nama Kasus Penggunaan</b>	Menjalankan Fungsi Next PowerPoint
<b>Nomor</b>	UC03
<b>Aktor</b>	Pengguna
<b>Kondisi Awal</b>	Program PowerPoint sudah terbuka
<b>Kondisi Akhir</b>	Sistem menjalankan fungsi next pada PowerPoint
<b>Alur Normal</b>	<ol style="list-style-type: none"><li>1. Pengguna melakukan permintaan pendeteksian suara</li><li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li><li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia</li><li>4. Sistem menerima masukan suara</li><li>5. Sistem menerjemahkan suara menjadi teks</li><li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li></ol>

**Tabel 3.1.8 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Next PowerPoint (2)**

<b>Alur Normal</b>	<p>7. Sistem mendeteksi perintah menjalankan fungsi next pada Powerpoint</p> <p>8. Sistem menjalankan fungsi next pada PowerPoint</p>
<b>Alur Alternatif</b>	<p>A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</p> <p>A1.1. Kembali ke alur normal nomor 3</p>

Berdasarkan skenario kasus penggunaan pada Tabel 3.1.5 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menjalankan fungsi next PowerPoint dapat dilihat pada Gambar 3.1.4



**Gambar 3.1.4 Diagram Aktivitas Menjalankan Fungsi Next PowerPoint**

### 3.1.3.6. Kasus Penggunaan Menjalankan Fungsi Previous PowerPoint

Pada kasus penggunaan menjalankan fungsi previous PowerPoint, pengguna dapat melakukan fungsi pada program melalui perintah suara Bahasa Indonesia berupa *previous* pada Powerpoint sehingga PowerPoint dapat memindahkan ke *slide* sebelumnya. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.7.

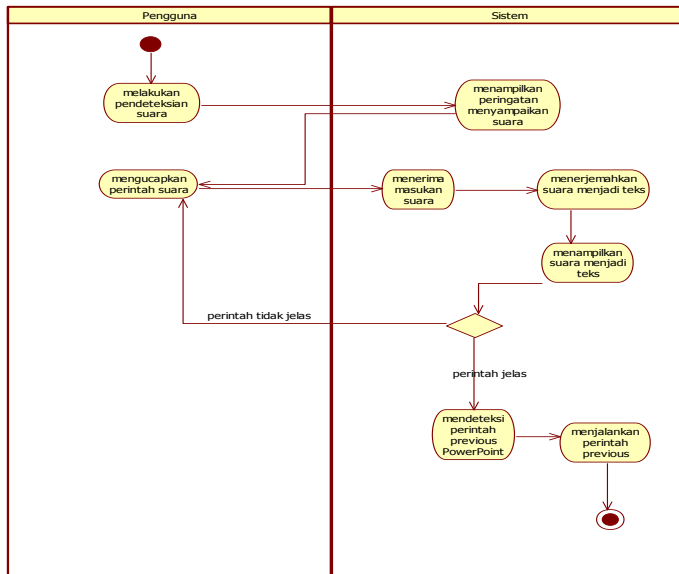
**Tabel 3.1.9 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Previous**

<b>Nama Kasus Penggunaan</b>	Menjalankan Fungsi Previous PowerPoint
<b>Nomor</b>	UC04
<b>Aktor</b>	Pengguna
<b>Kondisi Awal</b>	Program PowerPoint sudah terbuka
<b>Kondisi Akhir</b>	Sistem menjalankan fungsi previous pada PowerPoint
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia</li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li> </ol>

**Tabel 3.1.10 Rincian Alur Kasus Penggunaan Menjalankan Fungsi Previous PowerPoint (2)**

<b>Alur Normal</b>	<p>7. Sistem mendeteksi perintah menjalankan fungsi previous pada Powerpoint</p> <p>8. Sistem menjalankan fungsi previous pada PowerPoint</p>
<b>Alur Alternatif</b>	<p>A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</p> <p>A1.1. Kembali ke alur normal nomor 3</p>

Berdasarkan skenario kasus penggunaan pada Tabel 3.1.7 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menjalankan fungsi previous PowerPoint dapat dilihat pada Gambar 3.1.5.



**Gambar 3.1.5 Diagram Aktivitas Menjalankan Fungsi Previous PowerPoint**

### 3.1.3.7. Kasus Penggunaan Mematikan PowerPoint

Pada kasus penggunaan menutup PowerPoint, pengguna dapat menutup PowerPoint dengan perintah suara. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.11.

**Tabel 3.1.11 Rincian Alur Kasus Penggunaan Menutup PowerPoint**

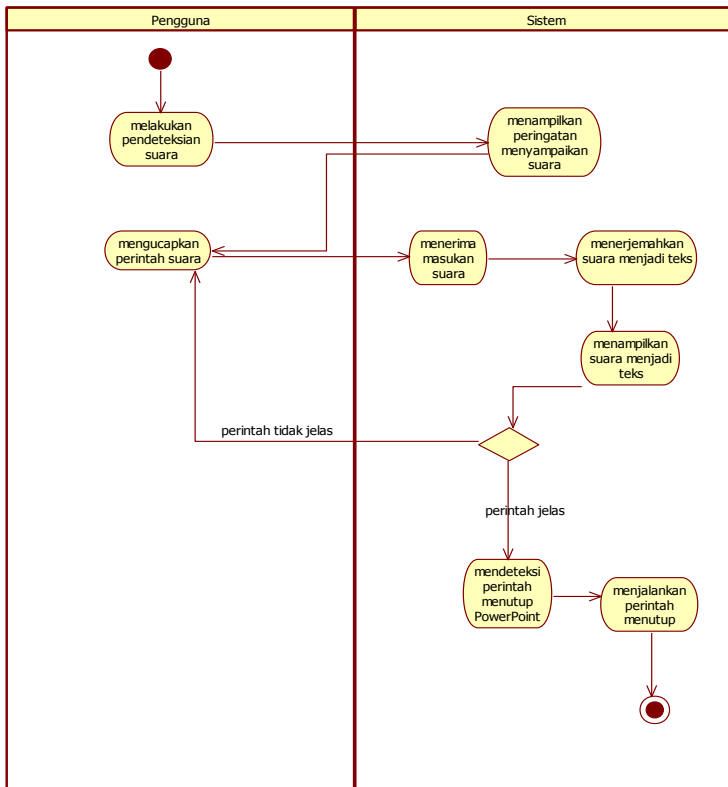
<b>Nama Kasus Penggunaan</b>	Menutup PowerPoint
<b>Nomor</b>	UC05
<b>Aktor</b>	Pengguna
<b>Kondisi Awal</b>	Program PowerPoint sudah terbuka
<b>Kondisi Akhir</b>	PowerPoint ditutup
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia</li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li> <li>7. Sistem mendeteksi perintah menjalankan tutup PowerPoint</li> <li>8. Sistem menutup PowerPoint</li> </ol>



**Tabel 3.1.12 Rincian Alur Kasus Penggunaan Menutup PowerPoint (2)**

<b>Alur Alternatif</b>	A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik A1.1. Kembali ke alur normal nomor 3
------------------------	---

Berdasarkan skenario kasus penggunaan pada Tabel 3.1.8 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menutup PowerPoint dapat dilihat pada Gambar 3.1.6.



**Gambar 3.1.6 Diagram Aktivitas Menutup PowerPoint**

### 3.1.3.8. Kasus Penggunaan Membuka Notepad

Pada kasus penggunaan membuka program, pengguna menekan tombol suara untuk memasukkan perintah membuka program kemudian komputer akan membuka program yang disebutkan.

Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.13.

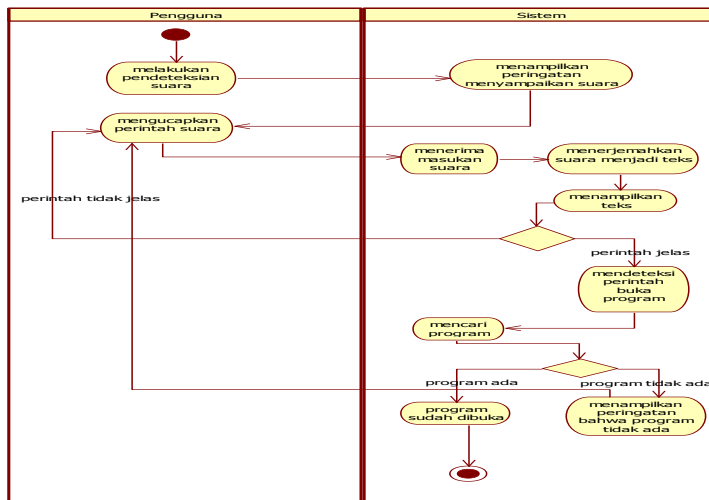
**Tabel 3.1.13 Rincian Alur Kasus Penggunaan Membuka Notepad**

<b>Nama Kasus Penggunaan</b>	Membuka Notepad
<b>Nomor</b>	UC06
<b>Aktor</b>	Pengguna
<b>Kondisi Awal</b>	Pengguna telah menekan tombol suara
<b>Kondisi Akhir</b>	Komputer membuka aplikasi notepad
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna mengucapkan perintah dalam Bahasa Indonesia</li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li> <li>7. Sistem mendeteksi perintah membuka program</li> </ol>

**Tabel 3.1.14 Rincian Alur Kasus Penggunaan Membuka Notepad (2)**

<b>Alur Normal</b>	<p>8. Sistem mencari program yang dimaksudkan</p> <p>A2. Program tidak ada</p> <p>9. Sistem membuka program</p>
<b>Alur Alternatif</b>	<p>A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</p> <p>A.1.1 Kembali ke alur normal nomor 3</p> <p>A2. Program tidak ada</p> <p>A.2.1 Sistem memberi peringatan program tidak ada</p> <p>A.2.2 Kembali ke alur normal nomor 8</p>

Berdasarkan skenario kasus penggunaan pada Tabel 3.1.9. Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan membuka program dapat dilihat pada Gambar 3.1.7.



**Gambar 3.1.7 Diagram Aktivitas Membuka Notepad**

### 3.1.3.9. Kasus Penggunaan Mematikan Komputer

Pada kasus penggunaan mematikan komputer, pengguna dapat mematikan komputer lewat suara tanpa menekan tombol mematikan komputer. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.1.10.

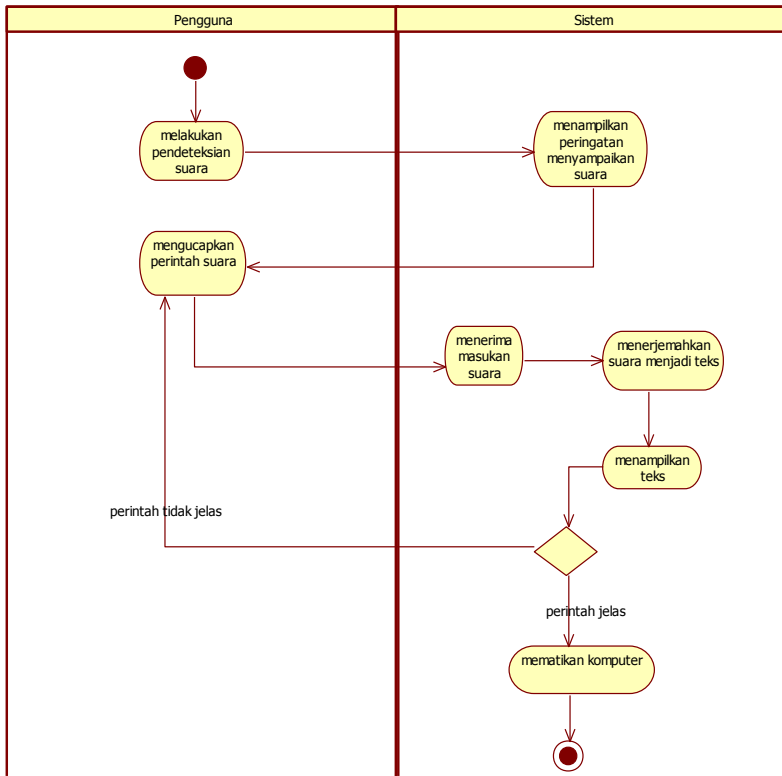
**Tabel 3.1.15 Rincian Alur Kasus Penggunaan Mematikan Komputer**

<b>Nama Kasus Penggunaan</b>	Mematikan Komputer
<b>Nomor</b>	UC07
<b>Aktor</b>	Pengguna
<b>Kondisi Awal</b>	Komputer sedang menyala
<b>Kondisi Akhir</b>	Komputer mati
<b>Alur Normal</b>	<ol style="list-style-type: none"> <li>1. Pengguna melakukan permintaan pendeteksian suara</li> <li>2. Sistem menampilkan peringatan untuk menyampaikan suara</li> <li>3. Pengguna menyampaikan perintah suara dalam Bahasa Indonesia</li> <li>4. Sistem menerima masukan suara</li> <li>5. Sistem menerjemahkan suara menjadi teks</li> <li>6. Sistem menampilkan teks A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik</li> <li>7. Sistem mendeteksi perintah mematikan komputer</li> <li>8. Sistem melakukan perintah matikan komputer</li> </ol>

**Tabel 3.1.16 Rincian Alur Kasus Penggunaan Mematikan Komputer (2)**

<b>Alur Alternatif</b>	A1. Perintah tidak jelas atau tidak dapat ditangkap dengan baik A1.1. Kembali ke alur normal nomor 3
------------------------	---

Berdasarkan skenario kasus penggunaan pada Tabel 3.1.10 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mematikan komputer dapat dilihat pada Gambar 3.1.8.



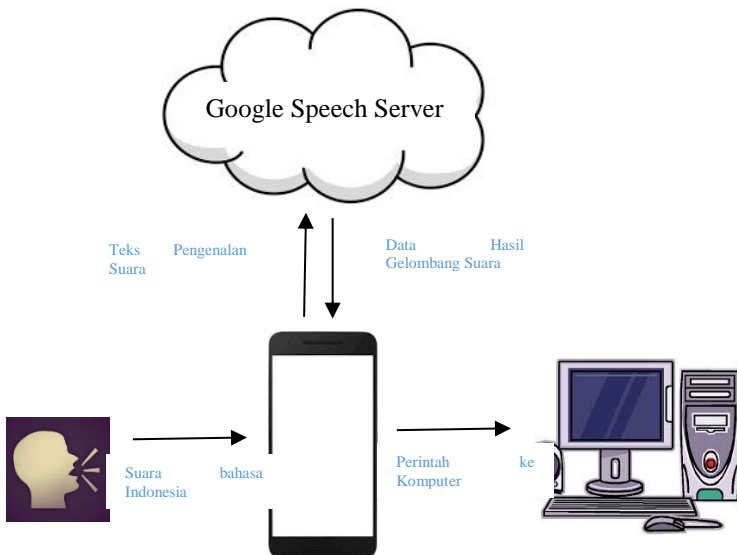
**Gambar 3.1.8 Diagram Aktivitas Mematikan Komputer**

### 3.2. Perancangan Sistem

Pada subbab ini dijelaskan mengenai tahapan perancangan sistem. Perancangan sistem ini dibagi menjadi beberapa bagian yang meliputi perancangan data, perancangan arsitektur sistem, perancangan proses aplikasi, dan perancangan antarmuka pengguna.

#### 3.2.1. Perancangan Arsitektur

Dalam tugas akhir ini akan dirancang dan diaplikasikan sebuah perangkat lunak menggunakan perangkat bergerak berbasis android dengan memanfaatkan masukan suara Bahasa Indonesia. Secara garis besar aplikasi ini memiliki rancangan arsitektur sistem yang dapat dilihat pada Gambar 3.4.



**Gambar 3.2.2 Rancangan Arsitektur Sistem Perintah Suara**

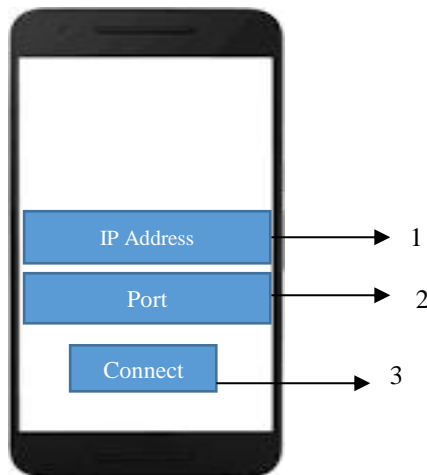
Dari gambar 3.4, dapat diketahui bahwa sistem beroperasi di antara aplikasi klien dan *server* dari Google. Aplikasi berjalan pada perangkat bergerak android dengan pendeteksian suara. Suara dikirim menjadi data gelombang suara oleh masukan suara android untuk dikirimkan ke *server* suara milik Google. *Server* ini kemudian memproses data gelombang suara menjadi kalimat atau teks dengan pilihan Bahasa Indonesia. Proses ini dilakukan dengan memanfaatkan API masukan suara pada perangkat bergerak android.

### 3.2.2. Perancangan Antarmuka

Pada bagian ini dijelaskan mengenai rancangan tampilan antarmuka pengguna dari sistem. Berikut akan dijelaskan masing-masing rancangan antarmuka

#### 3.2.2.1. Antarmuka Halaman Pembuka

Antarmuka halaman pembuka merupakan antarmuka yang ditampilkan ketika aplikasi pertama kali dijalankan. Rancangan antarmuka halaman pembuka dapat dilihat pada Gambar 3.9.



**Gambar 3.2.3 Rancangan Antarmuka Halaman Pembuka**

Pada rancangan antarmuka pada gambar 3.2.3 di atas berlatar belakang warna putih. Pada bagian antarmuka ini terdapat bagian-bagian antarmuka sebagai berikut:

1. IP Address

IP Address dimasukkan untuk membuka koneksi kepada *server* yang ada pada komputer. Bagian ini ditunjukkan pada gambar nomor 1.

2. Port

Untuk melakukan koneksi antar perangkat keras dibutuhkan port. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman pembuka dengan nomor 2.

3. Tombol *Connect*

Untuk memulai koneksi maka dibutuhkan sebuah tombol untuk memulai koneksi antar perangkat keras yang akan terhubung. Bagian ini ditunjukkan pada gambar rancangan antarmuka halaman pembuka dengan nomor 3.

### **3.2.3. Perancangan Proses**

Pada Subbab ini menjelaskan tentang perancangan proses-proses yang ada pada perangkat lunak. Berikut ini merupakan rancangan proses proses yang ada pada pengembangan Tugas Akhir ini.

#### **3.2.3.1. Proses Pada Kasus Penggunaan Membuka Berkas PowerPoint**

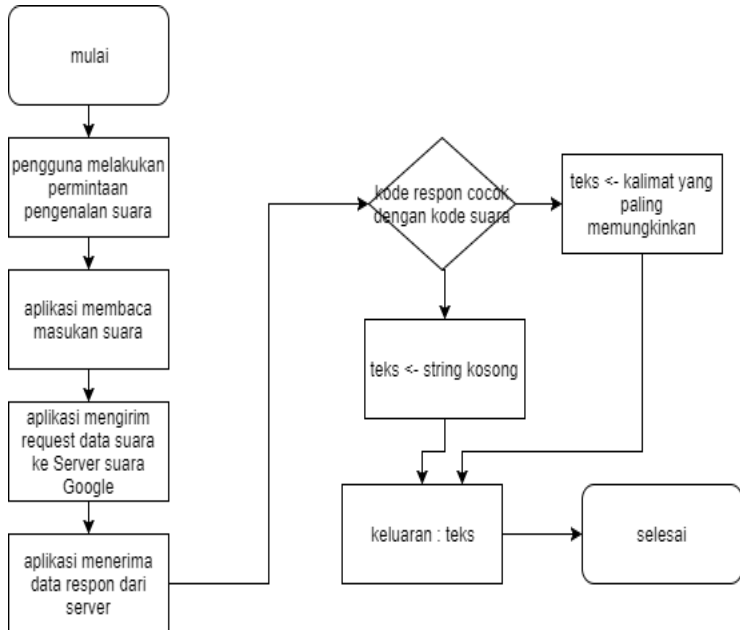
Merujuk pada spesifikasi kasus penggunaan membuka berkas PowerPoint. Terdapat beberapa proses yang dilakukan untuk memenuhi alur spesifikasi kasus penggunaan. Proses-proses tersebut antara lain proses deteksi suara dan pengecekan berkas

##### **3.2.3.1.1. Proses Deteksi Suara**

Proses deteksi suara merupakan proses pengenalan suara yang diterima aplikasi hingga menjadi teks. Proses deteksi suara pada aplikasi ini memanfaatkan masukan suara pada perangkat



bergerak berbasis android dan pengenal suara Google sebagai pemrosesnya. Diagram Alir proses deteksi suara dijelaskan pada gambar 3.2.5.



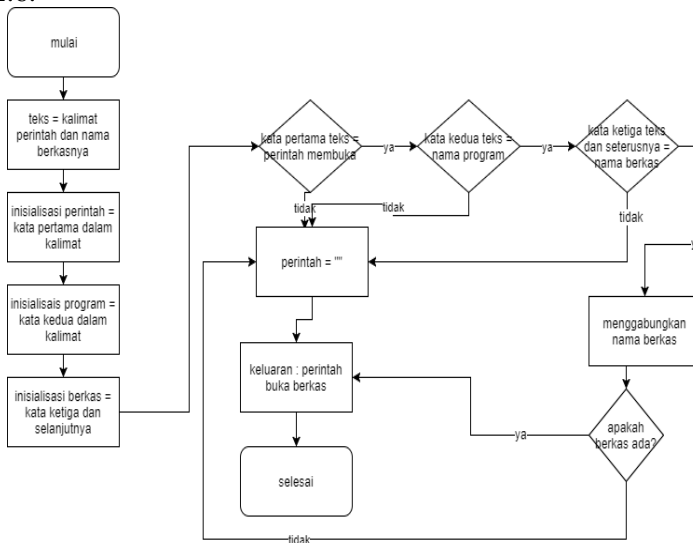
**Gambar 3.2.4 Diagram Alir Proses Deteksi Suara**

Proses deteksi suara dimulai dari permintaan pengguna untuk melakukan pengenalan suara. Kemudian aplikasi membaca masukan suara dari pengguna. Masukan dari proses ini merupakan suara yang ditangkap disekitar pengguna saat aplikasi melakukan pembacaan suara yang ditandai dengan munculnya tampilan pembacaan suara oleh aplikasi.

Aplikasi mengirim request data suara ke *server* suara Google dan kemudian menerima data respon dari *server*. Keluaran dari proses ini adalah teks.

### 3.2.3.1.2. Proses Pengambilan Kata Perintah Membuka Berkas

Proses pengambilan kata perintah merupakan lanjutan dari proses deteksi suara. Keluaran dari proses deteksi suara berupa teks. Kata perintah berada pada awal kalimat. Kata perintah disini yaitu untuk membuka berkas PowerPoint yang diinginkan. Tujuan dari proses ini adalah mendapatkan kata perintah membuka berkas. Proses ini dijelaskan pada gambar 3.2.6.



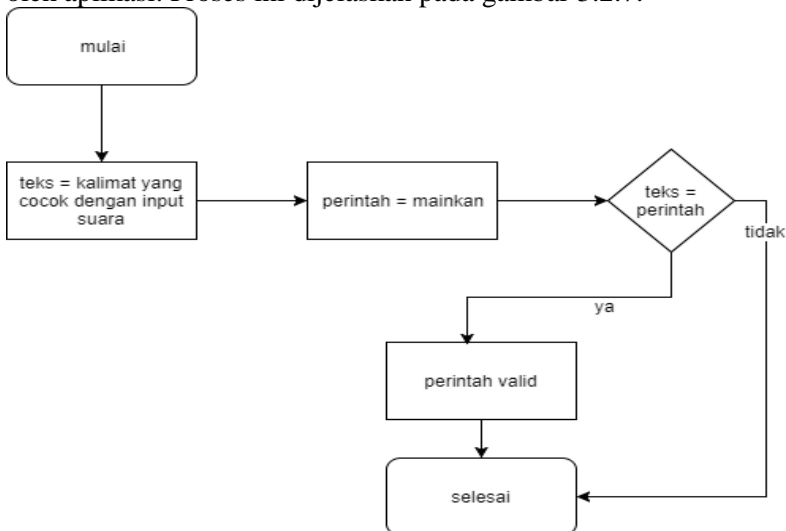
**Gambar 3.2.5 Diagram Alir Proses Pengambilan Kata Perintah Membuka Berkas**

Proses pengambilan kata perintah membuka berkas PowerPoint diawali dari pengucapan kata perintah. Kemudian dalam kalimat tersebut terdiri dari kata perintah, nama program PowerPoint, dan nama berkas. Ketika perintah yang terdapat pada kata pertama sama dengan membuka program maka dilanjutkan ke kata kedua. Jika kata kedua adalah aplikasi PowerPoint maka dilanjut ke kata ketiga. Pada kata ketiga dan seterusnya dilakukan penggabungan kata sehingga bilaman ada nama berkas yang dipisahkan oleh spasi, dapat dipanggil untuk dibuka. Kemudian dilakukan pengecekan berkas apakah berkas yang diinginkan ada

atau tidak. Keluaran dari proses ini adalah perintah membuka berkas PowerPoint.

### 3.2.3.2. Proses Pada Kasus Penggunaan Menjalankan Fungsi Play PowerPoint

Pada proses menjalankan fungsi play PowerPoint diawali dengan deteksi perintah suara yang dijelaskan pada proses sebelumnya. Kemudian perintah play PowerPoint akan dieksekusi oleh aplikasi. Proses ini dijelaskan pada gambar 3.2.7.



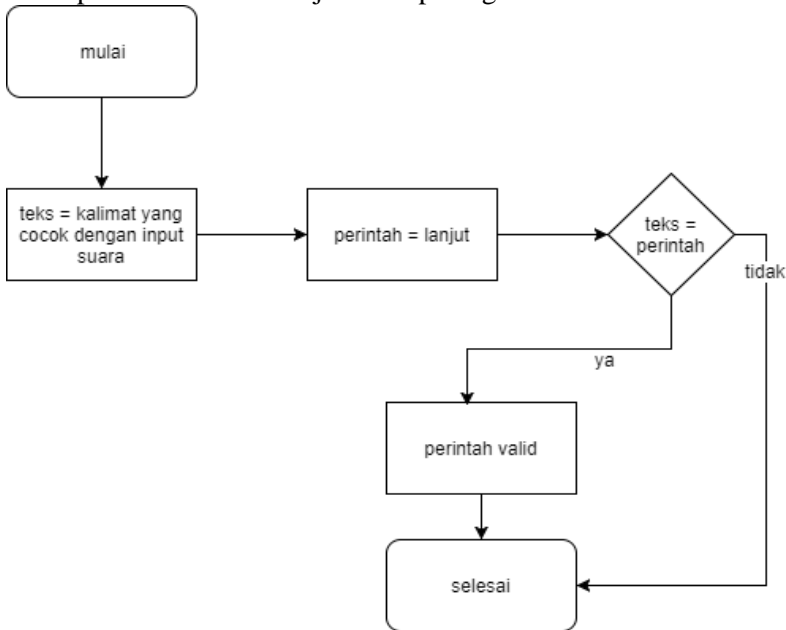
**Gambar 3.2.6 Diagram Alir Menjalankan Fungsi Play PowerPoint**

Pada proses menjalankan fungsi play PowerPoint, pengguna menggunakan 1 kata perintah. Bila fungsi play ingin dijalankan maka perintah yang diucapkan adalah “mainkan”. Jika perintah valid maka sistem akan menjalankan perintah play pada PowerPoint.

### 3.2.3.3. Proses Pada Kasus Penggunaan Menjalankan Fungsi Next PowerPoint

Pada proses menjalankan fungsi next PowerPoint diawali dengan deteksi perintah suara yang dijelaskan pada proses

sebelumnya. Kemudian perintah next PowerPoint akan dieksekusi oleh aplikasi. Proses ini dijelaskan pada gambar 3.2.8.

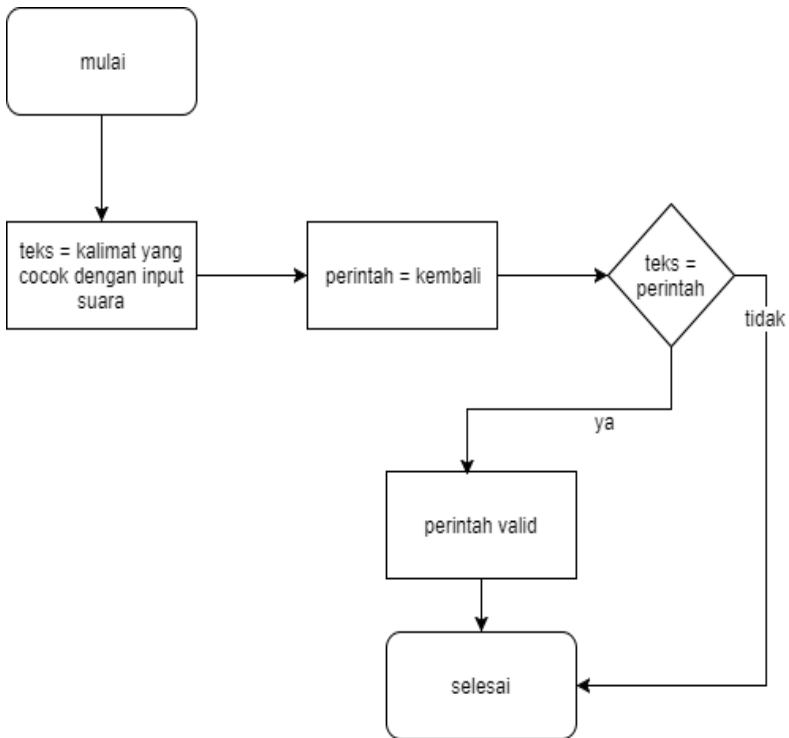


**Gambar 3.2.7 Diagram Alir Menjalankan Fungsi Next PowerPoint**

Pada proses menjalankan fungsi next PowerPoint, pengguna menggunakan 1 kata perintah. Bila fungsi next ingin dijalankan maka perintah yang diucapkan adalah “lanjut”. Jika perintah valid maka sistem akan menjalankan perintah next pada PowerPoint.

#### **3.2.3.4. Proses Pada Kasus Penggunaan Menjalankan Fungsi Previous PowerPoint**

Pada proses menjalankan fungsi previous PowerPoint diawali dengan deteksi perintah suara yang dijelaskan pada proses sebelumnya. Kemudian perintah previous PowerPoint akan dieksekusi oleh aplikasi. Proses ini dijelaskan pada gambar 3.2.9.

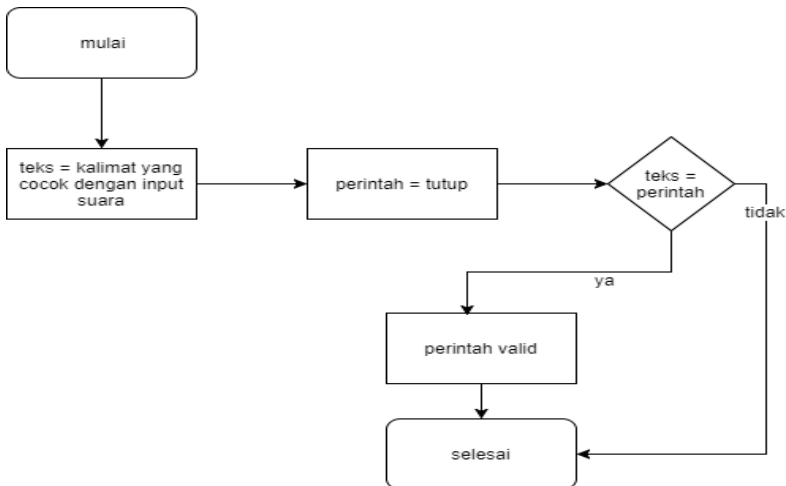


**Gambar 3.2.8 Diagram Alir Menjalankan Fungsi Previous PowerPoint**

Pada proses menjalankan fungsi previous PowerPoint, pengguna menggunakan 1 kata perintah. Bila fungsi previous ingin dijalankan maka perintah yang diucapkan adalah “kembali”. Jika perintah valid maka sistem akan menjalankan perintah previous pada PowerPoint.

### **3.2.3.5. Proses Pada Kasus Penggunaan Menutup PowerPoint**

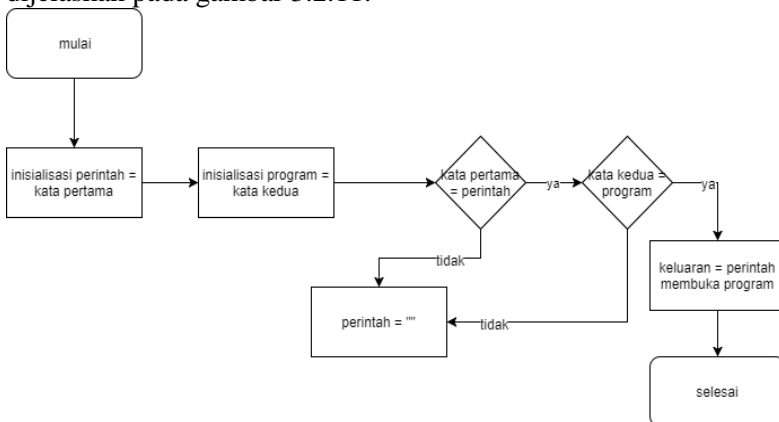
Pada proses menutup PowerPoint diawali dengan deteksi perintah suara yang dijelaskan pada proses sebelumnya. Kemudian perintah menutup PowerPoint akan dieksekusi oleh aplikasi. Proses ini dijelaskan pada gambar 3.2.10.



**Gambar 3.2.9 Diagram Alir Menutup PowerPoint**

### 3.2.3.6. Proses pada Kasus Penggunaan Membuka Notepad

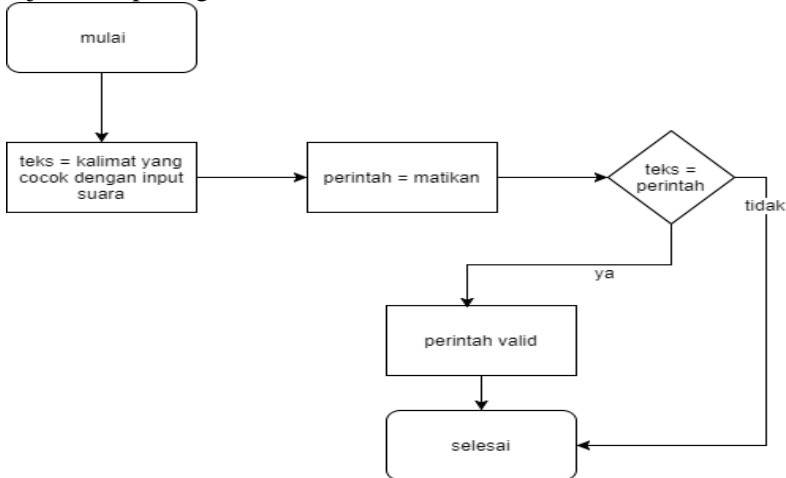
Pada proses membuka notepad diawali dengan deteksi perintah suara yang dijelaskan pada proses sebelumnya. Kemudian perintah membuka notepad akan dieksekusi oleh aplikasi. Proses ini dijelaskan pada gambar 3.2.11.



**Gambar 3.2.10 Diagram Alir Proses Membuka Notepad**

### 3.2.3.7. Proses Pada kasus Penggunaan Mematikan Komputer

Pada proses memmatikan komputer diawali dengan deteksi perintah suara yang dijelaskan pada proses sebelumnya. Kemudian perintah matikan komputer akan dieksekusi oleh aplikasi. Proses ini dijelaskan pada gambar 3.2.12.



**Gambar 3.2.11 Diagram Alir Mematikan Komputer**

### 3.2.4. Perancangan Daftar Perintah

Perintah pengoperasian komputer dalam hal ini dibagi menjadi tiga yaitu perintah navigasi aplikasi, perintah buka tutup aplikasi, dan perintah yang ada pada sistem komputer. Perintah-perintah ini merepresentasikan masing masing pengoperasian yang umum pada komputer.

#### 3.2.4.1. Perancangan Daftar Perintah Navigasi Aplikasi

Pada navigasi aplikasi, aplikasi yang akan diambil sebagai contoh adalah aplikasi PowerPoint. Aplikasi PowerPoint yang banyak digunakan dalam pengajaran di dalam kelas memiliki beberapa perintah umum yang sering dijalankan oleh pengajar.

Tabel 3.2.1 akan menjelaskan mengenai perintah navigasi yang ada pada aplikasi PowerPoint.

**Tabel 3.2.1 Daftar Perintah Navigasi Aplikasi PowerPoint**

No	Nama Perintah	Kegunaan
1.	Buka + presentasi + nama berkas dari program	Untuk membuka berkas PowerPoint
2.	Mainkan	Untuk memerintahkan PowerPoint dalam mode layar penuh
3.	Lanjut	Untuk memindahkan ke halaman sesudahnya pada PowerPoint
4.	Kembali	Untuk memindahkan ke halaman sebelumnya pada PowerPoint

Pada tabel 3.2.1 terdapat empat perintah navigasi pada PowerPoint yaitu Buka, Mainkan, Lanjut, dan Kembali. Buka merepresentasikan fungsi *open* pada PowerPoint. Perintah ini dapat langsung dijalankan dengan memanggil nama berkas PowerPoint yang diinginkan pada suatu folder tanpa harus mencari. Perintah Mainkan merupakan bentuk dari fungsi *play* dimana PowerPoint berada dalam mode layar penuh untuk menampilkan isi dari presentasi. Perintah Lanjut merupakan bentuk dari fungsi *next* pada PowerPoint sehingga bila pengguna atau pengajar ingin melanjutkan ke halaman selanjutnya bisa dengan memanggil perintah ini. Perintah Kembali merupakan bentuk dari fungsi *previous* yaitu mengembalikan PowerPoint ke halaman sebelumnya.

#### **3.2.4.2. Perancangan Daftar Perintah Buka Tutup Aplikasi**

Pada perancangan perintah buka tutup aplikasi, aplikasi yang akan diambil sebagai contoh adalah aplikasi notepad. Aplikasi notepad adalah aplikasi yang umum digunakan untuk melakukan pengetikan penanda karena sangat ringan dan sederhana. Aplikasi lain juga bisa dipanggil oleh perintah ini. Tabel 3.2.2 akan menjelaskan mengenai perintah buka tutup aplikasi yang ada pada komputer.



**Tabel 3.2.2 Daftar Perintah Buka Tutup Aplikasi**

No	Nama Perintah	Kegunaan
1.	Tutup + nama aplikasi	Untuk menutup aplikasi yang diinginkan
2.	Buka + nama aplikasi	Untuk membuka aplikasi tanpa ada nama berkas nya

Pada tabel 3.2.2 terdapat perintah membuka dan menutup aplikasi. Perintah Tutup merepresentasikan perintah *close app* pada komputer sehingga aplikasi yang diinginkan bisa ditutup oleh perintah ini. Perintah Buka merupakan bentuk dari fungsi *open app* pada komputer sehingga aplikasi yang diinginkan bisa dibuka oleh perintah ini.

### **3.2.4.3. Perancangan Daftar Perintah Sistem Komputer**

Pada perancangan perintah sistem komputer, perintah yang diambil adalah matikan komputer dan hidupkan ulang komputer. Matikan dan hidupkan ulang komputer merupakan bentuk perintah terhadap system komputer karena sudah disediakan oleh system operasi. Dua perintah ini umum digunakan dalam perkuliahan atau pengajaran setelah pengajar menjelaskan materi atau isi dari presentasi untuk para anak didik. Fungs ini bisa diakses pada panel awal tampilan antarmuka sistem operasi yang ada pada komputer. Tabel 3.2.3 akan menjelaskan mengenai perintah buka tutup aplikasi yang ada pada komputer.

**Tabel 3.2.3 Daftar Perintah Pada Aplikasi**

Nama Perintah	Kegunaan
Buka + nama program + nama berkas dari program	Untuk membuka berkas suatu program komputer
Mainkan	Untuk memerintahkan PowerPoint dalam mode layar penuh
Lanjut	Untuk memindahkan ke halaman sesudahnya pada PowerPoint
Kembali	Untuk memindahkan ke halaman sebelumnya pada PowerPoint

**Tabel 3.2.4 Daftar Perintah Pada Aplikasi (2)**

Tutup + nama program	Untuk menutup program yang diinginkan
Buka + nama program	Untuk membuka aplikasi tanpa ada nama berkas nya
Matikan	Perintah ini berfungsi untuk mematikan komputer

#### **3.2.4.4. Perancangan Daftar Nama Aplikasi**

Nama aplikasi yang dipanggil pada pengembangan aplikasi berbasis android ini bukan nama aplikasi yang sebenarnya. Bisa dikatakan pemanggilan aplikasi memakai nama alias yang umum dipakai saat pembelajaran dalam kelas.

**Tabel 3.2.5 Daftar Nama Alias Aplikasi**

No	Alias Aplikasi	Aplikasi yang Dibuka
1.	Presentasi	PowerPoint
2.	Notepad	Notepad
3.	Google	Browser Chrome

Pada tabel 3.2.4 dapat dilihat bahwa pemanggilan buka tutup aplikasi tidak menggunakan nama sebenarnya. Bentuk pemanggilan merupakan nama alias dari aplikasi tersebut yang umum pada saat pembelajaran.

## **BAB IV IMPLEMENTASI**

Bab ini membahas implementasi dari perancangan sistem sesuai dengan perancangan yang telah dibuat. Bahasa pemrograman yang digunakan untuk implementasi sistem adalah bahasa pemrograman Java dengan pustaka Apache Spark dan ImgScalr.

### **4.1. Lingkungan Implementasi**

Lingkungan implementasi sistem yang digunakan untuk mengembangkan tugas akhir memiliki spesifikasi perangkat keras dan perangkat lunak seperti ditampilkan pada Tabel 4.1.1.

**Tabel 4.1.1 Lingkungan Implementasi Sistem**

Perangkat	Spesifikasi
Perangkat keras	Prosesor: Intel(R) Celeron(R) CPU 1037U @ 1.80GHz (2 CPUs) , ~1.8GHz Memori: 8192MB Smartphone : Samsung Galaxy Core Duo Perangkat lain : USB Cable
Perangkat lunak	Sistem Operasi: Microsoft Windows 8.1 Enterprise 64-bit Perangkat Pengembang: Android Studio Perangkat Pembantu: Microsoft Word 2013
<i>Smartphone</i>	OS Android minimal versi 4.3
<i>Wifi</i>	<i>Wifi</i> yang ada di Teknik Informatika ITS dan sekitarnya
Ruangan	Laboratorium Teknik Informatika ITS Terdiri 5 orang : 2 orang saling berbicara 1 orang mendengarkan musik dengan speaker 2 orang bermain game dengan speaker

**Tabel 4.1.2 Lingkungan Implementasi Sistem (2)**

Berkas	- path berkas terletak pada drive F dengan nama folder coba - nama berkas tidak menggunakan karakter
Tempo Pengucapan	Normal

## 4.2. Implementasi Proses

Implementasi proses dilakukan berdasarkan perancangan proses yang sudah dijelaskan pada bab analisis dan perancangan.

### 4.2.1. Implementasi Tahap Deteksi Suara

Subbab ini membahas mengenai implementasi tahap deteksi suara. Pada tahap ini suara akan dipanggil melalui *library Intent* dari objek *RecognizerIntent*. *RecognizerIntent* kemudian membaca data suara dan dikirimkan ke *server* Google. Setiap kali *Activity* menerima respons dari luar, *Activity* mencocokkan dengan kode-kode permintaan yang dikirimkannya ke *server*. Jika didapatkan kode respon yang cocok dengan kode permintaan, maka teks diambil dari data yang paling cocok.

Proses dimulai dengan inisiasi objek *Intent* dari *RecognizerIntent*. Implementasi dengan kode semu dapat dilihat pada gambar

1	Intent <- objek RecognizerIntent baru
2	Tambah parameter untuk pendeteksian Bahasa Indonesia
3	Jalankan fungsi startActivityResult dengan parameter intent dan kode unik untuk pendeteksian suara

**Gambar 4.2.1 Kode Semu Untuk Inisiasi Objek *RecognizerIntent***

Ketika sudah didapatkan hasil respon dari *server* suara Google, maka *Intent* ini akan keluar dan membawa hasil yang bisa diakses melalui fungsi *onActivityResult()* pada kelas *Activity* yang memanggil *Intent*. Semua hasil pada *Intent* diterima oleh fungsi ini sehingga dilakukan pencocokan kode permintaan dengan kode

respon. Jika kode sesuai maka dimunculkan hasil pembacaan teks dari permintaan kode suara tadi.

Implementasi kode semu dari pengolahan respon dari *server* menjadi teks diperlihatkan pada gambar 4.2.2. Implementasi lainnya dapat dilihat pada lampiran B – kode sumber.

1	requestCode <- kode respon dari server Google
2	Data <- isi respon dari <i>server</i> Google
3	If requestCode = kode unik untuk pendeteksian suara Matches <- list kata yang cocok dari data If Matches tidak kosong Statement <- isi matches yang pertama Else Statement <- string kosong Tampilkan statement

**Gambar 4.2.2 Kode Semu Pengolahan Suara Menjadi Teks dari *Server***

#### 4.2.2. Implementasi Lalu Lintas Data Teks ke Server

Pengiriman data teks dari perangkat *smartphone* ke *server* dilakukan dengan mengkoneksikan perangkat *smartphone* ke *server* melalui wifi yang ada di gedung Teknik Informatika. Kode sumber pengiriman data teks ke *server* dapat dilihat pada kode sumber 4.2.3.

1	data <- teks dari <i>server</i> Google
2	Socket <- pustaka socket dengan parameter IP Address dan port
3	Data dikirim ke <i>server</i> komputer
4	Socket <- flush()
5	Socket <- close()

**Gambar 4.2.3 Kode Semu Pengiriman dari Perangkat Bergerak ke *Server* Komputer**

Setelah dilakukan pengiriman data teks ke *server*, maka *server* menerima data tersebut. Data yang diterima berupa string agar bisa dibaca oleh *server* dan diterjemahkan dengan benar. Proses tersebut dapat ditunjukkan pada gambar 4.2.4.

1	clientSocket <- menerima pengiriman socket dengan parameter port
2	Pesan <- dijadikan string agar bisa dibaca
3	Tampilkan pesan

**Gambar 4.2.4 Kode Semu Penerimaan Perintah dari Perangkat Bergerak ke *Server* Komputer**

**4.2.3. Implementasi Membuka Berkas PowerPoint**

Membuka berkas PowerPoint memiliki parameter letak folder yang berkasnya akan dibuka. Setelah itu dilakukan pengecekan apakah berkasnya ada atau tidak. Jika ada maka berkas PowerPoint dibuka, jika tidak ada maka tidak melakukan apa-apa. Implementasi ini dapat dilihat pada lampiran B – kode sumber. Untuk implementasi pada membuka Berkas PowerPoint dapat dilihat pada kode semu pada gambar.

1	If perintah = buka
2	If perintah = presentasi
3	Pesan setelah perintah dijoinkan untuk mendapat nama berkas File <- data nama berkas If File sama dengan nama berkas Buka berkas powerpoint If File tidak ada Tampilkan tidak ada berkas

**4.2.4. Implementasi Menjalankan Fungsi Play PowerPoint**

Menjalankan fungsi play PowerPoint menggunakan pustaka *Robot*. *Robot* digunakan untuk melakukan pemencetan pada keyboard. Untuk fungsi play dilakukan pemencetan F5. Implementasi proses ini dapat dilihat pada lampiran B – kode sumber.

**4.2.5. Implementasi Menjalankan Fungsi Next PowerPoint**

Menjalankan fungsi next PowerPoint menggunakan pustaka *Robot*. *Robot* digunakan untuk melakukan pemencetan pada keyboard. Untuk fungsi next dilakukan pemencetan *VK\_DOWN*.

Implementasi proses ini dapat dilihat pada lampiran B – kode sumber.

#### **4.2.6. Implementasi Menjalankan Fungsi Previous PowerPoint**

Menjalankan fungsi previous PowerPoint menggunakan pustaka *Robot*. *Robot* digunakan untuk melakukan pemencetan pada keyboard. Untuk fungsi previous dilakukan pemencetan *VK\_UP*. Implementasi proses ini dapat dilihat pada lampiran B – kode sumber.

#### **4.2.7. Implementasi Menutup PowerPoint**

Implementasi menutup PowerPoint menggunakan pustaka *Runtime* dengan objek *Process*. Jadi semua proses yang sedang berjalan pada komputer dimasukkan ke dalam list. Kemudian dicek apakah ada nama PowerPoint dalam list proses tersebut. Jika ada maka langsung dimatikan. Implementasi ini dapat dilihat pada lampiran B – kode sumber.

#### **4.2.8. Implementasi Membuka Notepad**

Implementasi membuka notepad menggunakan pustaka *Runtime* dengan objek *Process*. Program notepad dieksekusi oleh pustaka *runtime*. Implementasi untuk proses ini dapat dilihat pada lampiran B – kode sumber.

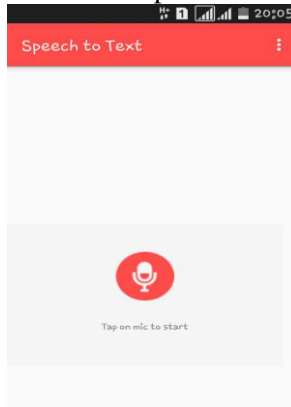
#### **4.2.9. Implementasi Mematikan Komputer**

Pada tahap ini akan dilakukan eksekusi perintah untuk mematikan komputer. Sama dengan implementasi sebelumnya, untuk mematikan komputer digunakan pustaka *runtime*. *Runtime* yang berisi kode mematikan komputer, dieksekusi oleh *server* sehingga menyebabkan komputer melakukan proses *shut down*.

### **4.3. Implementasi Antarmuka Perangkat Lunak**

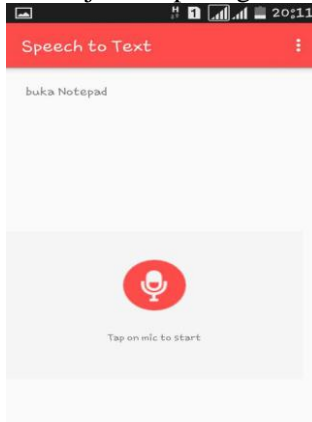
Implementasi tampilan antarmuka pengguna pada perangkat lunak berbasis *android*. Pada antarmuka ini terdapat judul perangkat lunak. Bagian berwarna merah bundar merupakan tombol bicara yang sudah dijelaskan pada rancangan antarmuka

halaman utama yang akan digunakan oleh pengguna untuk melakukan masukan suara. Kemudian pada antarmuka ini akan muncul teks keluaran dari pembangkitan proses suara sehingga setelah pengguna melakukan masukan suara maka hasil masukan suara tersebut bisa muncul pada antarmuka ini. Kemudian dari antarmuka ini, hasilnya akan dikirim pada antarmuka lain.



**Gambar 4.3.1** Gambar Implementasi Antarmuka Pengguna *smartphone*

Gambar 4.3.1 menunjukkan halaman pertama yang akan dilihat pengguna. Seperti ditunjukkan pada gambar4.3.2.



**Gambar 4.3.2** Gambar Implementasi Pengubahan Suara ke Teks



## **BAB V**

### **PENGUJIAN DAN EVALUASI**

Bab ini membahas uji coba dan evaluasi terhadap perangkat lunak yang telah dikembangkan dari implementasi aplikasi pendeteksi perintah suara berbahasa Indonesia berbasis android untuk mengoperasikan komputer.

#### **5.1. Lingkungan Pengujian**

Lingkungan pengujian sistem pada pengerjaan Tugas Akhir ini dilakukan pada lingkungan dan alat kakas sebagai berikut:

- a. *Server*
  - Prosesor : Intel® Celeron® CPU 1037U @ 1.80GHz (2 CPUs) , ~1.8GHz
  - RAM : 8192 MB
  - Jenis *Device* : *Netbook*
  - Sistem Operasi : Microsoft Windows 8.1 Enterprise
- b. *Smartphone*
  - Prosesor : Quad-core 1.2 GHz
  - Jenis *Device* : *Smartphone*
  - Sistem Operasi : Android Kitkat 4.4.2
- c. Wifi : Wifi yang ada di Teknik Informatika ITS
- d. Ruangan : Laboratorium Teknik Informatika ITS
  - Jumlah : 5 orang
    - 1 orang saling berbicara
    - 1 orang mendengarkan musik dengan speaker
    - 1 orang bermain game dengan speaker
- e. Berkas :
  - Dokumen tersimpan pada drive F dengan nama folder coba
  - Nama berkas tidak menggunakan karakter
- f. Tempo Pengucapan : Normal

## 5.2. Skenario Uji Coba

Pada subbab ini akan dijelaskan mengenai skenario uji coba yang telah dilakukan. Terdapat beberapa skenario uji coba yang telah dilakukan, diantaranya yaitu:

1. Skenario pengujian 1: Membuka Berkas PowerPoint.
2. Skenario pengujian 2: Menjalankan Fungsi Play PowerPoint
3. Skenario pengujian 3 : Menjalankan Fungsi Next PowerPoint
4. Skenario pengujian 4 : menjalankan Fungsi Previous PowerPoint
5. Skenario pengujian 5 : Menutup PowerPoint
6. Skenario pengujian 6 : Membuka Notepad
7. Skenario pengujian 7 : Mematikan Komputer.

## 5.3. Skenario Pengujian 1: Membuka Berkas PowerPoint

Pengujian membuka file PowerPoint merupakan pengujian terhadap kemampuan program untuk menerima perintah suara yang diberikan oleh pengguna, menganalisa, dan mengirim perintah ke *server* komputer. Pengujian ini dilakukan saat pengguna akan membuka file PowerPoint. Rincian skenario pada kasus penggunaan ini bisa dilihat pada tabel 5.3.1.

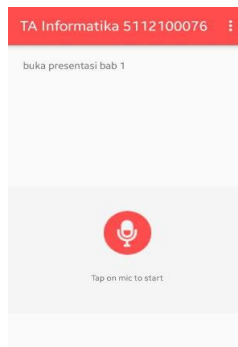
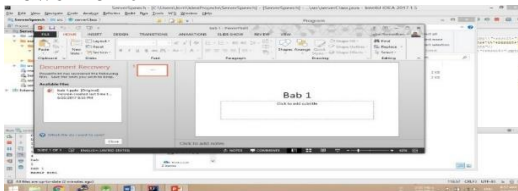
**Tabel 5.3.1 Pengujian Membuka Berkas PowerPoint**

<b>Referensi Penggunaan</b>	<b>Kasus</b>	UC01
<b>Nama</b>	Pengujian membuka berkas PowerPoint	
<b>Tujuan Pengujian</b>	Membuka berkas PowerPoint dengan perintah suara	
<b>Skenario 1</b>	<b>Pengguna membuka file PowerPoint yang diinginkan</b>	
<b>Kondisi Awal</b>	Pengguna berada pada antarmuka komunikasi	

**Tabel 5.3.2 Pengujian Membuka Berkas PowerPoint (2)**

<b>Data Uji</b>	Isi Perintah : buka presentasi bab 1
<b>Langkah Pengujian</b>	Pengguna mengucapkan perintah suara “buka presentasi bab 1”
<b>Hasil yang Diharapkan</b>	Program akan menampilkan file PowerPoint dengan nama file bab 1
<b>Hasil yang Didapatkan</b>	Program menampilkan file PowerPoint dengan nama berkas bab 1
<b>Hasil Pengujian</b>	Berhasil
<b>Kondisi Akhir</b>	Berkas PowerPoint dengan nama bab 1 terbuka

Gambar 5.3.1 menunjukkan hasil dari skenario pengujian pada tabel 5.3.1. Pada gambar tersebut terlihat bahwa pengucapan perintah “membuka presentasi bab 1” akan membuka aplikasi PowerPoint dengan nama bab 1 yang ditunjukkan pada gambar 5.3.2.

**Gambar 5.3.1 Hasil Pengujian Skenario 1 Membuka Berkas PowerPoint****Gambar 5.3.2 Kondisi Akhir Skenario 1 Membuka Aplikasi si PowerPoint**

Hasil pemanggilan Kalimat “Buka Presentasi Bab 1” dengan 20 kali pemanggilan pada ruangan berisi 5 orang. Hasil Pemanggilan perintah dapat dilihat pada Tabel 5.8.2.

**Tabel 5.3.3 Hasil Pemanggilan Perintah Buka Presentasi**

Buka Presentasi Bab 1	15
Google Presentasi Bab 1	1
Bukan Presentasi Bab 1	2
Buka Prestasi Bab 1	2

#### **5.4. Skenario Pengujian 2: Menjalankan Fungsi Play PowerPoint**

Pengujian menjalankan fungsi play PowerPoint merupakan pengujian terhadap program untuk melakukan perintah play pada PowerPoint. Perintah play pada PowerPoint diganti dengan kata perintah “mainkan”. Rincian scenario pengujian pada kasus penggunaan ini dapat dilihat pada tabel 5.4.1.

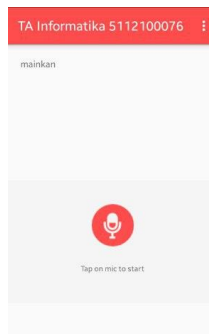
**Tabel 5.4.1 Pengujian Menjalankan Fungsi Play PowerPoint**

<b>Referensi Kasus Penggunaan</b>	UC02
<b>Nama</b>	Pengujian menjalankan fungsi Powerpoint
<b>Tujuan Pengujian</b>	Menjalankan fungsi PowerPoint dengan perintah suara
<b>Skenario 1</b>	<b>Pengguna menjalankan fungsi PowerPoint : play</b>
<b>Kondisi Awal</b>	Pengguna berada pada antarmuka komunikasi
<b>Data Uji</b>	Isi Perintah : mainkan
<b>Langkah Pengujian</b>	Pengguna mengucapkan perintah suara “mainkan”
<b>Hasil yang Diharapkan</b>	Program akan memainkan PowerPoint dalam mode fullscreen

**Tabel 5.4.2 Pengujian Menjalankan Fungsi Play PowerPoint (2)**

<b>Hasil yang Didapatkan</b>	Program memainkan PowerPoint dalam mode fullscreen
<b>Hasil Pengujian</b>	Berhasil
<b>Kondisi Akhir</b>	PowerPoint dalam mode <i>play</i> fullscreen

Gambar 5.4.1 merupakan hasil dari pengujian skenario 1 menjalankan fungsi “play”. Pada gambar tersebut terlihat perintah “mainkan” yang berarti “play” pada aplikasi PowerPoint.

**Gambar 5.4.1 Pengujian Skenario 1 Menjalankan Fungsi “play”**

## Bab 1

**Gambar 5.4.2 Hasil Pengujian Skenario 1 Menjalankan Fungsi “play”**

Hasil pemanggilan Kalimat “Mainkan” dengan 20 kali pemanggilan pada ruangan berisi 5 orang. Hasil Pemanggilan perintah dapat dilihat pada Tabel 5.8.2.

**Tabel 5.4.3 Hasil Pemanggilan Perintah Mainkan**

Mainkan	20
---------	----

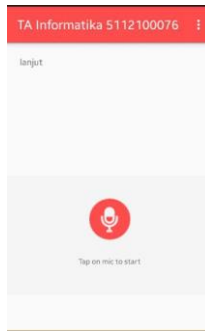
### 5.5. Skenario Pengujian 3 : Menjalankan Fungsi Next PowerPoint

Pengujian menjalankan fungsi next PowerPoint merupakan pengujian terhadap program untuk melakukan perintah next pada PowerPoint. Perintah next pada PowerPoint diganti dengan kata perintah “lanjut”. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada tabel 5.5.1.

**Tabel 5.5.1 Pengujian Menjalankan Fungsi Next PowerPoint**

<b>Referensi Kasus Penggunaan</b>	UC03
<b>Skenario 1</b>	<b>Pengguna menjalankan fungsi PowerPoint : next</b>
<b>Kondisi Awal</b>	Pengguna berada pada mode <i>play fullscreen</i>
<b>Data Uji</b>	Isi perintah : lanjut
<b>Langkah Pengujian</b>	Pengguna mengucapkan perintah suara “lanjut”
<b>Hasil yang Diharapkan</b>	Program akan mengubah <i>slide</i> PowerPoint saat ini ke <i>slide</i> PowerPoint selanjutnya
<b>Hasil yang Didapatkan</b>	Program mengubah <i>slide</i> PowerPoint saat ini ke <i>slide</i> PowerPoint selanjutnya
<b>Hasil Pengujian</b>	Berhasil
<b>Kondisi Akhir</b>	<i>Slide</i> PowerPoint berubah ke <i>slide</i> selanjutnya

Gambar 5.5.1 merupakan hasil dari pengujian skenario 1 menjalankan fungsi “next”. Pada gambar tersebut terlihat perintah “lanjut” yang berarti “next” pada aplikasi PowerPoint.



**Gambar 5.5.1 Pengujian Skenario 2 Menjalankan Fungsi “next”**

Hasil pemanggilan Kalimat “Lanjut” dengan 20 kali pemanggilan pada ruangan berisi 5 orang. Hasil Pemanggilan perintah dapat dilihat pada Tabel 5.8.2.

**Tabel 5.5.2 Hasil Pemanggilan Perintah Lanjut**

Lanjut	20
--------	----

## 5.6. Skenario Pengujian 4 : Menjalankan Fungsi Previous PowerPoint

Pengujian menjalankan fungsi previous PowerPoint merupakan pengujian terhadap program untuk melakukan perintah previous pada PowerPoint. Perintah previous pada PowerPoint diganti dengan kata perintah “kembali”. Rincian skenario pengujian pada kasus penggunaan ini dapat dilihat pada tabel 5.6.1.

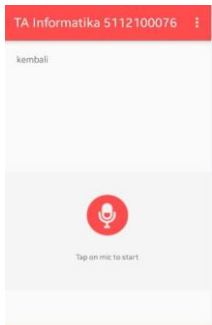
**Tabel 5.6.1 Pengujian Menjalankan Fungsi Previous PowerPoint**

<b>Referensi Kasus Penggunaan</b>	UC04
<b>Skenario 1</b>	<b>Pengguna menjalankan fungsi PowerPoint : previous</b>
<b>Kondisi Awal</b>	Pengguna berada pada mode <i>play</i> fullscreen
<b>Data Uji</b>	Isi perintah : kembali

**Tabel 5.6.2 Pengujian Menjalankan Fungsi Previous PowerPoint (2)**

<b>Langkah Pengujian</b>	Pengguna mengucapkan perintah suara “kembali”
<b>Hasil yang Diharapkan</b>	Program akan mengubah <i>slide</i> PowerPoint saat ini ke <i>slide</i> PowerPoint sebelumnya
<b>Hasil yang Didapatkan</b>	Program mengubah <i>slide</i> PowerPoint saat ini ke <i>slide</i> PowerPoint sebelumnya
<b>Hasil Pengujian</b>	Berhasil
<b>Kondisi Akhir</b>	<i>Slide</i> PowerPoint berubah ke <i>slide</i> sebelumnya

Sedangkan untuk gambar 5.6.1 merupakan hasil dari pengujian scenario 1 menjalankan fungsi “previous”. Perintah “previous” dalam Bahasa Inggris diubah menjadi “kembali” dalam Bahasa Indonesia. Kemudian pada gambar 5.6.1 terlihat bahwa perintah “kembali” merupakan hasil dari pengujian scenario 1 yaitu “previous” dalam aplikasi PowerPoint.



**Gambar 5.6.1 Pengujian Skenario 3 Menjalankan fungsi “previous”**

Hasil pemanggilan Kalimat “Kembali” dengan 20 kali pemanggilan pada ruangan berisi 5 orang. Pengujian terhadap 5 orang tersebut memiliki peran tersendiri seperti yang sudah disebutkan pada bab implementasi dan bab pengujian. Hasil Pemanggilan perintah dapat dilihat pada Tabel 5.8.2.

**Tabel 5.6.3 Hasil Pemanggilan Perintah Kembali**

Kembali	20
---------	----



### 5.7. Skenario Pengujian 5 : Menutup PowerPoint

Pengujian menutup PowerPoint merupakan pengujian terhadap kemampuan program untuk menutup PowerPoint. Rincian skenario dapat dilihat pada tabel 5.7.1.

**Tabel 5.7.1 Pengujian Menutup PowerPoint**

<b>Referensi Penggunaan</b>	<b>Kasus</b>	UC05
<b>Nama</b>		Menutup PowerPoint
<b>Tujuan Pengujian</b>		Untuk menutup PowerPoint
<b>Skenario 1</b>		<b>Pengguna menutup PowerPoint</b>
<b>Kondisi Awal</b>		Pengguna berada pada antarmuka komunikasi
<b>Data Uji</b>		Isi Perintah : tutup presentasi
<b>Langkah Pengujian</b>		Pengguna mengucapkan perintah suara “tutup presentasi”
<b>Hasil yang Diharapkan</b>		Program akan menutup PowerPoint
<b>Hasil yang Didapatkan</b>		Program menutup PowerPoint
<b>Hasil Pengujian</b>		Berhasil
<b>Kondisi Akhir</b>		Program PowerPoint tertutup

Pada gambar 5.7.1 menunjukkan hasil pengujian skenario 1 mematikan komputer. Perintah “tutup presentasi” akan dikirim ke *server* yang terdapat pada komputer sehingga komputer bisa melakukan perintah tutup PowerPoint melalui suara.



**Gambar 5.7.1 Pengujian Menutup PowerPoint**

Hasil pemanggilan Kalimat “Menutup Presentasi” dengan 20 kali pemanggilan pada ruangan berisi 5 orang. Hasil Pemanggilan perintah dapat dilihat pada Tabel 5.8.2.

**Tabel 5.7.2 Hasil Pemanggilan Perintah Tutup Presentasi**

Tutup Presentasi	18
Tutup Prestasi	2

### 5.8. Skenario Pengujian 6: Membuka Notepad

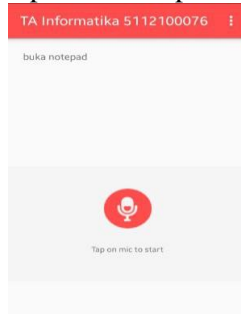
Pengujian membuka file PowerPoint merupakan pengujian terhadap kemampuan program untuk menerima perintah suara yang diberikan oleh pengguna, menganalisa, dan mengirim perintah ke *server* komputer. Rincian skenario pengujian pada kasus penggunaan ini bisa dilihat pada tabel 5.8.1.

**Tabel 5.8.1 Pengujian Membuka Notepad**

<b>Referensi Kasus Penggunaan</b>	UC06
<b>Nama</b>	Membuka notepad
<b>Tujuan Pengujian</b>	Untuk membuka notepad
<b>Skenario 1</b>	<b>Pengguna membuka notepad</b>
<b>Kondisi Awal</b>	Pengguna berada pada antarmuka komunikasi
<b>Data Uji</b>	Isi Perintah : buka notepad
<b>Langkah Pengujian</b>	Pengguna mengucapkan perintah suara “buka notepad”
<b>Hasil yang Diharapkan</b>	Program akan membuka notepad
<b>Hasil yang Didapatkan</b>	Program membuka notepad
<b>Hasil Pengujian</b>	Berhasil
<b>Kondisi Akhir</b>	Program notepad terbuka

Pada gambar 5.8.1 menunjukkan hasil pengujian skenario 1 membuka notepad. Perintah “buka notepad” akan dikirim ke

*server* yang terdapat pada komputer sehingga komputer bisa melakukan membuka notepad melalui perintah suara.



**Gambar 5.8.1 Hasil Pengujian Skenario 1 Membuka Notepad**

Hasil pemanggilan Kalimat Buka Notepad dengan 20x pada ruangan berisi 5 orang. Hasil Pemanggilan perintah dapat dilihat pada Tabel 5.8.2.

**Tabel 5.8.2 Hasil Pemanggilan Perintah Buka Notepad**

Buka notepad	15
Bukan otak	3
Bukan tepat	2

## 5.9. Skenario Pengujian 7: Mematikan Komputer

Pengujian membuka file PowerPoint merupakan pengujian terhadap kemampuan program untuk menerima perintah suara yang diberikan oleh pengguna, menganalisa, dan mengirim perintah ke *server* komputer. Rincian skenario pengujian pada kasus penggunaan ini bisa dilihat pada tabel 5.8.1.

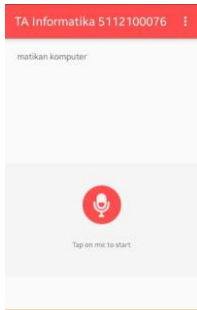
**Tabel 5.9.1 Pengujian Mematikan Komputer**

<b>Referensi Penggunaan</b>	<b>Kasus</b>	UC07
<b>Nama</b>	Pengujian mematikan komputer	

Tabel 5.9.2 Pengujian Mematikan Komputer (2)

Tujuan Pengujian	Mematikan komputer
Skenario 1	Pengguna mematikan komputer
Kondisi Awal	Pengguna berada pada antarmuka komunikasi
Data Uji	Isi Perintah : matikan komputer
Langkah Pengujian	Pengguna mengucapkan perintah suara “matikan komputer”
Hasil yang Diharapkan	Program akan mematikan komputer
Hasil yang Didapatkan	Program mematikan komputer
Hasil Pengujian	Berhasil
Kondisi Akhir	Komputer dalam keadaan mati

Gambar 5.6.1 merupakan hasil pengujian skenario 1 membuka notepad. Perintah “matikan komputer” dijalankan melalui *server* yang ada pada komputer dan kemudian komputer akan mati dengan sendirinya.



Gambar 5.9.1 Hasil Pengujian Skenario 1 Mematikan Komputer

Pada skenario ini dilakukan uji coba mematikan komputer dengan perintah suara. Perintah yang diberikan adalah : “Matikan Komputer”.

Tabel 5.9.3 Hasil Pemanggilan Perintah Matikan Komputer

Matikan Komputer	20
------------------	----

### 5.10. Pengujian Kebergunaan

Pengujian kebergunaan dilakukan untuk mengetahui tingkat kebergunaan program terhadap pengguna. Kebergunaan yang diuji meliputi kemudahan dalam pengoperasian program serta kemudahan dalam memberikan perintah suara pada program.

Pengujian dilakukan dengan memberi kesempatan pada pengguna untuk mencoba sendiri program yang telah dikembangkan. Uji coba yang dilakukan oleh pengguna meliputi, membuka file Powerpoint, menjalankan fungsi PowerPoint, mematikan komputer, dan membuka notepad. Rincian kuesioner dapat dilihat pada tabel 5.7.1.

Daftar pengguna yang melakukan pengujian dapat dilihat pada tabel 5.7.2. Pemilihan pengguna didasarkan pada usia pengguna sehingga bisa mencakup beberapa golongan. Hasil dari kuesioner dapat dilihat pada tabel 5.7.3. Hasil dari kuesioner disajikan dalam bentuk prosentase pilihan jawaban dari masing-masing pertanyaan yang diajukan ke pengguna.

**Tabel 5.10.1 Kuesioner Pengujian Kebergunaan**

No.	Pernyataan	Pilihan Jawaban			
		Sangat Setuju	Setuju	Tidak Setuju	Sangat Tidak Setuju
1.	Menurut saya, aplikasi ini mudah digunakan				
2.	Menurut saya, tidak sulit memberikan perintah suara pada program				
3.	Menurut saya, tidak sulit dalam membuka berkas PowerPoint dengan perintah suara				

**Tabel 5.10.2 Kuesioner Pengujian Kebergunaan (2)**

<b>No.</b>	<b>Pernyataan</b>	<b>Pilihan Jawaban</b>			
4.	Menurut saya, tidak sulit dalam menjalankan fungsi PowerPoint				
5.	Menurut saya, tidak sulit dalam mematikan komputer menggunakan perintah suara				
6.	Menurut saya, tidak sulit dalam membuka notepad dengan perintah suara				
7.	Menurut saya, saya terbantu dengan program ini				

**Tabel 5.10.3 Daftar Penguji Program**

<b>No.</b>	<b>Nama</b>	<b>Perangkat</b>	<b>Sistem Operasi</b>	<b>Alamat</b>
1.	Mohammad Iqbal Ramadhan	Samsung Galaxy Core Duos	Android Kitkat 4.4	Lumajang
2.	M. Sholahuddin	Oppo Neo 7	Android Lollipop 5.1	Jombang
3.	Abdullah Mujib	Redmi 4a	Android Marshmallow 6.0.1	Surabaya
4.	Irfan Naufal	Samsung Galaxy Tab A	OS Lollipop 5.1.1	Surabaya

**Tabel 5.10.4 Daftar Penguji Program (2)**

5.	Aditya	Xiaomi Redmi 2	Android Marshmallow 6.0.1	Malang
6.	Ihsan Prasetya	Xiaomi Mi 4x	Android Marshmallow 6.0.1	Padang
7.	Achmad Saiful	Xiaomi Redmi note 3 pro	Android Marshmallow 6.0.1	Surabaya
8.	Hendro	Asus Zenfone 2 Deluxe	OS Marshmallow 6.0.1	Semarang
9.	Wahyu Hidayat	Asus Zenfone 5	Android Lollipop 5.0	Solo
10.	Adim	Samsung Galaxy J2 Prime	Android Marshmallow 6.0	Solo
11.	Aldhiaz	Xiaomi Redmi Note 3	Android Lollipop 5.0.2	Cirebon
12.	Aldo	Asus Zenfone 4	Android Lollipop 5.0.2	Malang

**Tabel 5.10.5 Hasil Kuesioner**

No.	Pernyataan	Pilihan Jawaban			
		Sangat Setuju	Setuju	Tidak Setuju	Sangat Tidak Setuju
1.	Menurut saya, aplikasi ini mudah digunakan	10%	70%	20%	-

**Tabel 5.10.6 Hasil Kuesioner (2)**

2.	Menurut saya, tidak sulit memberikan perintah suara pada program	-	80%	20%	-
3.	Menurut saya, tidak sulit dalam membuka file PowerPoint dengan perintah suara	20%	70%	10%	-
4.	Menurut saya, tidak sulit dalam menjalankan fungsi PowerPoint	50%	50%	-	-
5.	Menurut saya, tidak sulit dalam mematikan komputer menggunakan perintah suara	100%	-	-	-
6.	Menurut saya, tidak sulit dalam membuka notepad dengan perintah suara	40%	60%	-	-
7.	Menurut saya, saya terbantu dengan program ini	-	40%	40%	20%

### 5.11. Evaluasi Pengujian

Pada subbab ini akan diberikan hasil evaluasi dari pengujian-pengujian yang telah dilakukan. Evaluasi yang diberikan meliputi evaluasi pengujian fungsionalitas dan evaluasi pengujian kebergunaan.



### 5.11.1. Evaluasi Pengujian Fungsionalitas

Rangkuman mengenai evaluasi pengujian fungsionalitas dapat dilihat pada tabel. Dari semua data tersebut dapat dilihat bahwa semua skenario pengujian berhasil dan program berjalan dengan baik. Sehingga bisa disimpulkan bahwa fungsionalitas program ini bekerja dengan baik.

**Tabel 5.11.1 Rangkuman Hasil Pengujian**

Kasus Penggunaan	Nama	Skenario	Hasil
UC01	Membuka Berkas PowerPoint	Skenario 1	Berhasil
UC02	Menjalankan Fungsi Play PowerPoint	Skenario 1	Berhasil
UC03	Menjalankan Fungsi Next PowerPoint	Skenario 1	Berhasil
UC04	Menjalankan Fungsi Previous PowerPoint	Skenario 1	Berhasil
UC05	Menutup PowerPoint	Skenario 1	Berhasil
UC06	Membuka notepad	Skenario 1	Berhasil
UC07	Mematikan Komputer	Skenario 1	Berhasil

### 5.11.2. Evaluasi Pengujian Kebergunaan

Berdasarkan pada tabel 5.7.3 dapat ditarik kesimpulan bahwa program mudah digunakan. Hal ini dapat dilihat pada rata-rata 70% pengguna setuju bahwa aplikasi mudah digunakan dan 40% terbantu dengan aplikasi ini. Hanya saja, rata-rata 60% pengguna tidak merasa terbantu dengan aplikasi ini karena untuk beberapa fitur masih lebih cepat

menggunakan secara manual. Rata-rata pengguna terbantu pada saat perintah membuka program karena cepat tanpa perlu membuka folder.

## **BAB VI**

### **KESIMPULAN DAN SARAN**

Pada bab ini akan diberikan kesimpulan yang diperoleh selama pengerjaan Tugas Akhir dan saran mengenai pengembangan yang dapat dilakukan terhadap Tugas Akhir ini di masa yang akan datang.

#### **6.1. Kesimpulan**

Dari hasil pengamatan selama proses perancangan, implementasi, dan pengujian perangkat lunak yang dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Pustaka *Intent* dari *RecognizerIntent* digunakan untuk pengenalan suara pada aplikasi ini. Pustaka ini dapat diakses melalui API suara Google. Pengembangan dan penerapan pustaka ke dalam aplikasi untuk membangkitkan pengenalan suara dari perangkat bergerak berhasil dilakukan pada aplikasi ini.
2. Aplikasi berbasis android dan *server* menggunakan java bisa dihubungkan dengan jaringan wi-fi. Dari perangkat bergerak, hasil pengubahan perintah suara berbahasa Indonesia ke teks dikirim ke *server* yang ada pada komputer untuk kemudian dieksekusi perintah apa yang diinginkan. Proses ini berhasil diatur pada aplikasi ini.
3. Interaksi antara pengguna dan program untuk menjalankan perintah yang diberikan oleh pengguna telah berhasil diatur dengan mendefinisikan kata perintah dan kata pelengkapanya.

#### **6.2. Saran**

Berikut merupakan beberapa saran untuk pengembangan sistem di masa yang akan datang. Saran-saran ini didasarkan pada hasil perancangan, implementasi dan pengujian yang telah dilakukan.

1. Penambahan kecerdasan buatan agar program mampu mengenali kebiasaan pengucapan atau perintah yang diberikan pengguna kepada program.
2. Menggunakan pustaka pengenalan suara yang tidak membutuhkan koneksi internet agar pengenalan suara bisa responsive
3. Penambahan variasi kalimat perintah yang digunakan sehingga bisa terjadi interaksi dua arah.
4. Lingkungan percobaan yang lebih terkendali baik dalam kuantitas maupun kualitas.

## DAFTAR PUSTAKA

- [1] Maulana Putra, Fajar. 2013. Rancang Bangun Perangkat Lunak Pengirim SMS Dengan Perintah Suara Berbahasa Indonesia Pada Platform Android. Surabaya : Penerbitan ITS.
- [2] Najib, Muhammad. 2013. Rancang Bangun Aplikasi Penuntun Jalan Dengan Perintah Suara Berbahasa Indonesia Pada Perangkat Bergerak Berbasis Android. Surabaya : Penerbitan ITS.
- [3] Tenriawaru Prajatami, Andi. 2012. Pengaruh Pembelajaran Presentasi PowerPoint Terhadap Pemahaman Siswa Kelas XI. Diambil dari : <http://www.academia.edu/6756513> [Online]. Diakses : 18 Juli 2017.
- [4] Naila Fauzia, Rahmani. 2014. Pengembangan Media Interaktif PowerPoint Pembelajaran Wayang Untuk Siswa SMP Kelas VIII D.I. Yogyakarta. Diambil dari : <http://eprints.uny.ac.id/18112/> [Online]. Diakses : 18 Juli 2017.
- [5] Norhafizah, "Voice Control of Home Appliances using Android," 978-1-4799-6947-0/14/\$31.00 ©2014 IEEE.
- [6] Mulhern, Nicholas, "Designing Android Applications using Voice Controlled Command," 978-0-7695-4964-4/13/\$26.00 ©2013 IEEE.
- [7] Khadilkar, Shraddav Uddhav, "Android phone controlled Voice, Gesture, Touch screen operated Smart Wheelchair," 978-1-4673-1142-7/12/\$31.00 ©2012 IEEE.
- [8] Bareta, Stephanie, "Designing Android Applications with Both Online and Offline Voice Control of Household Devices," 978-1-4799-6272-3/15/\$31.00 ©2015 IEEE.
- [9] Pengertian wifi dan fungsinya. <http://www.indraservicelaptop.com/2014/04/pengertian-wifi-dan-fungsinya.html>. 2011. Diakses 18 Juli 2017

- [10] Pengertian dan Fungsi Wifi.  
<https://wifi.id/news/157/Pengertian+dan+Fungsi+WiFi>. 2016.  
Diakses 18 Juli 2017.
- [11] Pustaka Runtime Java.  
[https://www.tutorialspoint.com/java/lang/java\\_lang\\_runtime.htm](https://www.tutorialspoint.com/java/lang/java_lang_runtime.htm).  
2017. Diakses 18 Juli 2017.

## Lampiran A – Kode Layout

```
<?xml version="1.0" encoding="utf-8"?>
<menu
  xmlns:android="http://schemas.android.com/apk/res/android">
  <item
    android:id="@+id/action_settings"
    android:orderInCategory="100"
    android:showAsAction="never"
```

**Kode Sumber 1 activity.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<menu
  xmlns:android="http://schemas.android.com/apk/res/android">
  <item
    android:id="@+id/action_settings"
    android:orderInCategory="100"
    android:showAsAction="never"
```

**Kode Sumber 2 Colors.xml**

*{Halaman ini sengaja dikosongkan}*



## Lampiran B – Kode Sumber

```
private class SendMessage extends
AsyncTask<Void, Void, Void> {

    private String sendMessage;

    public SendMessage(String message){
        this.sendMessage = message;
    }

    @Override
    protected Void doInBackground(Void...
params) {
        try {
            //Log.i("speech", "send_msg : "
+ sendMessage);
            client = new Socket(ipAddress,
Integer.parseInt(port));
            printwriter = new
PrintWriter(client.getOutputStream(),
true);

            printwriter.write(sendMessage);
            printwriter.flush();
            printwriter.close();
            client.close();
        } catch (UnknownHostException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        return null;
    }
}
```

**Kode Sumber 3 Kode Sumber Mengirim Pesan Perintah**

```

@Override
protected void onActivityResult(int
requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode,
resultCode, data);

    switch (requestCode) {
        case REQ_CODE_SPEECH_INPUT: {
            if (resultCode == RESULT_OK &&
null != data) {
                ArrayList<String> result =
data.getStringArrayListExtra(RecognizerIntent.E
XTRA_RESULTS);
                //Log.i("speech", "result :" +
result.get(0));
                this.setMessage(result.get(0));

mVoiceInputTv.setText(result.get(0));
                //Log.i("speech",
"result_msg :" + message);
                sendMessageTask = new
SendMessage(message);
                sendMessageTask.execute();
            }
            break;
        }
    }
}

```

Kode Sumber 4 Kode Sumber Pengecekan Kode

```

private void startVoiceInput() {
    Intent intent = new
    Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH
    );

    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE
    _MODEL, "id-ID");

    //intent.putExtra(RecognizerIntent.EXTRA_LANGUA
    GE_MODEL,
    RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);

    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE
    , Locale.getDefault());

    intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
    "Sebutkan perintah anda");
    try {
        startActivityResult(intent,
        REQ_CODE_SPEECH_INPUT);
    } catch (ActivityNotFoundException a) {

    }
}

```

**Kode Sumber 5 Kode Sumber Deteksi Suara**

```

public static boolean isProcessRunning(String
serviceName) throws Exception {

    Process p = Runtime.getRuntime().exec(TASKLIST);
    BufferedReader reader = new BufferedReader(new
InputStreamReader(
        p.getInputStream()));
    String line;
    while ((line = reader.readLine()) != null) {

        //System.out.println(line);
        if (line.contains(serviceName)) {
            return true;
        }
    }

    return false;
}

public static void killProcess(String serviceName)
throws Exception {

    Runtime.getRuntime().exec(KILL + serviceName);
}

```

### Kode Sumber 6 Kode Sumber Menutup Program

```

clientSocket = serverSocket.accept(); //accept
client connection
inputStreamReader = new
InputStreamReader(clientSocket.getInputStream()
);
bufferedReader = new
BufferedReader(inputStreamReader);
pesan = bufferedReader.readLine();
System.out.println(pesan);
String kalimat = pesan.toLowerCase();

```

### Kode Sumber 7 Kode Sumber Menerima Koneksi Perangkat Bergerak

```

if (tangkapPerintah[angka].equals("buka")) {
    System.out.println(angka);
    angka++;
    try {
        if
(tangkapPerintah[angka].equals("presentasi")) {
            System.out.println(angka);
            System.out.println(hasil);
            StringBuilder builder = new
StringBuilder();
            for(int i = 2; i < hasil; i++){

System.out.println(tangkapPerintah[i]);
                builder.append(tangkapPerintah[i]);
                if (i < hasil-1){
                    builder.append(" ");
                }
            }
            String result = builder.toString();
            System.out.println(result);
            Runtime runtime = Runtime.getRuntime();
            String file =
folderData+"\\ "+result+".pptx";
            File f = new File(file);
            if(f.exists() && !f.isDirectory()){
                String[] s = new String[]{"C:\\Program
Files\\Microsoft Office\\Office15\\POWERPNT.exe",
folderData+"\\ "+result+".pptx"};
                runtime.exec(s);
            }
            else {
                System.out.println("Berkas Tidak
Ditemukan");
            }

```

**Kode Sumber 8 Kode Sumber Membuka PowerPoint**

```

else if (tangkapPerintah[angka].equals("mainkan")) {
    Robot robot = null;
    try {
        robot = new Robot();
    } catch (AWTException e) {
        e.printStackTrace();
    }
    robot.keyPress(KeyEvent.VK_F5);
    robot.keyRelease(KeyEvent.VK_F5);
}
else if (tangkapPerintah[angka].equals("lanjut")) {
    Robot robot = null;
    try {
        robot = new Robot();
    } catch (AWTException r) {
        r.printStackTrace();
    }
    robot.keyPress(KeyEvent.VK_DOWN);
    robot.keyRelease(KeyEvent.VK_DOWN);
}
else if (tangkapPerintah[angka].equals("kembali")) {
    Robot robot = null;
    try {
        robot = new Robot();
    } catch (AWTException e) {
        e.printStackTrace();
    }
    robot.keyPress(KeyEvent.VK_UP);
    robot.keyRelease(KeyEvent.VK_UP);
}

```

```

else if (tangkapPerintah[angka].equals("matikan")) {
    Runtime runtime = Runtime.getRuntime();
    Process proc = runtime.exec("shutdown -s -t
0");
}

```

### Kode Sumber 10 Kode Sumber Mematikan Komputer

```

else if (tangkapPerintah[angka].equals("tutup")) {
    angka++;
    if (tangkapPerintah[angka].equals("presentasi"))
    {
        String matiProgram = "POWERPNT.EXE";
        try {
            if (isProcessRunning(matiProgram)) {
                killProcess(matiProgram);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

### Kode Sumber 11 Kode Sumber Menutup Program

*{Halaman ini sengaja dikosongkan}*



## **BIODATA PENULIS**



Mohammad Iqbal Ramadhan, anak pertama dari satu bersaudara yang lahir di Lumajang pada tanggal 25 Februari 1994. Penulis telah menempuh pendidikan formal mulai dari SD Islam Timpokersan Lumajang (2000-2006), SMP Negeri 1 Lumajang (2006-2009), SMA Negeri 2 Lumajang (2009-2012) dan terakhir sebagai mahasiswa Jurusan Teknik Informatika Institut Teknologi Sepuluh

Nopember dengan rumpun mata kuliah Manajemen Informasi (2012-2017). Lulus dari SMA penulis melanjutkan pendidikan di jurusan teknik informatika. Semasa kuliah penulis aktif mengikuti berbagai kepanitiaan diantaranya Schematics 2013, Schematics 2014, Kegiatan Kepemanduan ITS dan kegiatan lain di jurusan. Penulis juga aktif menjadi anggota Himpunan Mahasiswa Teknik Computer-Informatika pada tahun 2013 dan 2014. Penulis dapat dihubungi melalui email [rmiqbalramadhan@gmail.com](mailto:rmiqbalramadhan@gmail.com)

*{Halaman ini sengaja dikosongkan}*